# Proof of Stake

Mikael Beyene

**Zusammenfassung**

Durch das Mining, welches bei Kryptowährungen, die auf Proof of Work setzen zum Einsatz kommt, wurden im Jahre 2017 alleine bei Bitcoin 29 TWh[1] verbraucht. Neue Ansätze versprechen die Gewährleistung der Funktionsweise ohne Mining und dem damit verbundenen Stromverbrauch. Durch die Abkehr vom Mining als Grundbaustein für die Sicherheit von Kryptowährungen ergeben sich neue Chancen, aber auch neue Herausforderungen, beim Design und Risiken im Einsatz. Ziel dieser Arbeit ist die Vorstellung einer Alternative zu Proof of Work, welche korrektes Verhalten der Protokollteilnehmer durch ein pfandähnliches Konzept motiviert sowie vereinzelte Gegenüberstellungen des neuen, Proof of Stake genannten, Verfahrens mit Proof of Work. Des Weiteren die Vorstellung von Angriffen in diesem Umfeld, Mechanismen zur Verhinderung und, wo möglich, das Ziehen von Vergleichen zu Proof of Work.

## 1 Einführung

Im Folgenden soll eine Motivation für Proof of Stake und der Kontext der Anwendung durchleuchtet werden.

### 1.1 Bitcoin

Das revolutionäre am, im Jahre 2008 erschienen, Paper *Bitcoin: A Peer-to-Peer Electronic Cash System* ist nicht die Anwendung von digitalen Signaturen für den eindeutigen Besitz von Münzen gewesen, sondern der Gebrauch von teuren Berechnungen, genannt *Proof of Work (PoW)*, zum Protokollieren eines korrekten Transaktionsverlaufes. D.h. insbesondere die Verhinderung des sog. *Double Spending*[1] – erstmals ohne das Hinzuziehen von vertrauenswürdigen Drittparteien [Naka08, S. 1, S. 8]. Das Durchführen dieser Proof-of-Work-Berechnungen wird auch *Mining* genannt.

### 1.2 Proof of Work und Energieverbrauch

Das Konzept des Proof of Works wurde als Kostenfunktion erstmals von Adam Back in *Hashcash* als Gegenmaßnahme zu *Denial-of-Service* Attacken vorgeschlagen. Wichtige Eigenschaften waren dabei eine parametrisierbar teure Kostenfunktion für die Rechnungen und das effiziente Verifizieren des Ergebnisses [Ba$^+$ot02, S.1].

Die teuren Berechnungen führten 2017, alleine bei Bitcoin, zu einem Energieverbrauch von 29 TWh[2]. Diese Problematik wurde früh erkannt und so gab es schon 2011 erste Vorschläge, die darauf abzielten das Mining lediglich zu simulieren[3].

[1] Weiteres zu Double Spending in Abschnitt 5.4

[2] Siehe `https://powercompare.co.uk/bitcoin/`

[3] Siehe auch *Proof of stake instead of proof of work*, Beitrag von Nutzer QuatumMechanic und andere, `https://bitcointalk.org/index.php?topic=27787.0`

In Proof-of-Work-Protokollen erhält der erste Teilnehmer, welcher eine akzeptierte Lösung der Kostenfunktion berechnet, dadurch implizit das Recht den nächsten Block vorzuschlagen und bekommt den Block Reward. Auf diese Weise wird der nächste Block, und der dazugehörige Block Reward (bzw. das Recht den nächsten Block zu minen) probabilistisch und proportional zur Rechenleistung zugeteilt. Dieser Prozess ist inhärent *zufällig* und bedarf keiner externen Entropie [KRDO17, S. 2f.].

*Proof of Stake (PoS)* genannte Protokolle simulieren das Mining, indem das Recht auf den nächsten Block probabilistisch und proportional zu einer Art Pfand zugeteilt wird. Dieses Pfand wird von den Minern, bzw. deren Pendants, die in einem Proof-of-Stake-Kontext *Validatoren* genannt werden, gestellt.

# 2 Begriffsklärung

Dieser Abschnitt befasst sich mit der Definition einiger Begriffe, die für das Verständnis dieser Arbeit wichtig sind. Weitere Begriffe werden bei der Verwendung definiert oder ergeben sich aus dem Kontext.

## 2.1 Staking

*Stake* bezeichnet das Pfand, welches *Validatoren* bezahlen. Kennzeichnend für dieses Stake ist, dass es in der Kryptowährung des Systems, das durch das Proof-of-Stake-Verfahren gesichert werden soll, bezahlt wird [BuGr17, S. 1]. Dieser Vorgang kann als *Staking* bezeichnet werden. Es lässt sich ein Henne-Ei-Problem erkennen.

Das *Bootstrapping* von Proof-of-Stake-Systemen ist im Gegensatz zu Proof-of-Work-Systemen nicht trivial [BeGM16, S. 2]. Miner können ihre Hardware zum Minen benutzen, Kryptowährung, die zum *Staken* benötigt wird, ist initial jedoch noch nicht verteilt. Auf dieses *Bootstrapping Problem* wird in Abschnitt 5.2.2 näher eingegangen.

## 2.2 Slots und Epochen

Slots sind diskrete Einheiten von Zeit. Ein Block ist immer genau einem Slot zugeordnet. Eine feste Sequenz von Slots kann zu einer Epoche zusammengefasst werden. [KRDO17, S. 3ff.]

## 2.3 Forks und Branches

Forks und Branches der Blockchain sind analog zu Bäumen in der theoretischen Informatik definiert. Zur Verdeutlichen zeigt Abbildung 1 ein Beispiel aus Git.

Zu sehen sind eine Verkettung von *Commits*, diese sollen Blöcken entsprechen. Nach Block $C2$ gab es einen Fork. Es gibt nun zwei Branches die mit $C4$ und $C5$ enden.

Jedem Slot wird in der Regel ein Block zugeordnet, wenn keine Branches existieren. Die Ausnahmen bilden im Beispiel hier $C4$ und $C3$.

[4]Scott Chacon, Ben Straub. Pro Git. Abbildung 20. `https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging`, aufgerufen am 27. Feburuar 2018

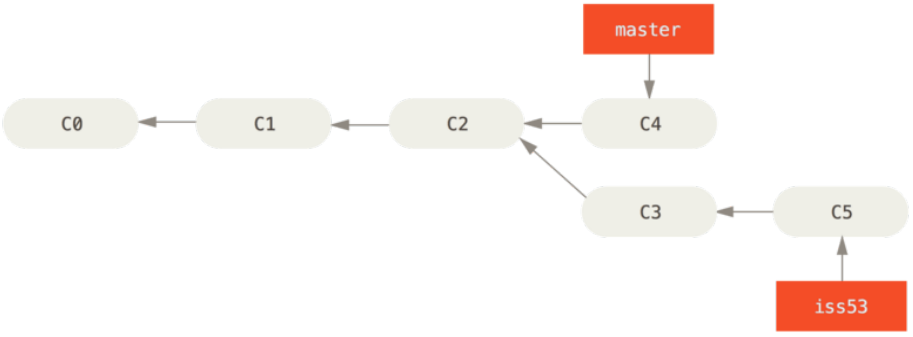


Abbildung 1: Branches am Beispiel von Git[4]

### 2.4 Angriff

Ein Angriff ist der Versuch eine alternative Blockchain, bzw. einen alternativen Branch, bereitzustellen, die aufgrund von Metriken wie z.B. der *Longest Chain* Regel, von den anderen Teilnehmern im Netzwerk akzeptiert wird. Oder ein Versuch die Funktionsweise dieser Metriken zu stören.

## 3 Grundlagen

Proof of Stake mag im Kontext von Blockchains und Kryptowährungen zwar neu sein, die grundlegenden Konzepte unterscheiden sich jedoch wenig von schon vorhandener Forschung (z.B. im Bereich von verteilten Systemen). Auch gibt es viele Parallelen zum herkömmlichen Proof-of-Work-Ansatz.

### 3.1 Ablauf

Der Ablauf eines Proof-of-Stake-Protokolls lässt sich folgendermaßen unterteilen:

1. **Staken**
   Dies findet meist mit einer sog. *Bond Transaction* statt. Durch das Wegsperren des Stakes wird der Menge der Validatoren beigetreten.
2. **Leader Election Process**
   Nach dem Staken wird der Validator bestimmt. Dafür gibt es verschiedene Ansätze. Dies trifft insbesondere beim *chain-based Proof-of-Stake*-Ansatz aus Abschnitt 4.1 zu, welcher das Mining simuliert.
3. **Anreizgesteuertes Verhalten**
   Durch das Koppeln von positiven und negativen Anreizen an bestimmte Aktionen kann das Verhalten von rationalen Validatoren gesteuert werden.

### 3.2 Leader Election Process

Die in Abschnitt 1.2 erwähnte Zuteilung des nächsten Blocks, welche mit einer Wahrscheinlichkeit stattfindet, die proportional zur Rechenleistung des Miners ist, wird auch *Leader*

*Election Process* genannt. Diese abstrakte Bezeichnung lässt die Anwedung im PoS-Kontext zu, dort wird der *Leader* probabilistisch anhand seines Stakes aus der Menge der Validatoren gewählt.

Manche Ansätze wählen die Belegung der Validatoren zu Beginn einer Epoche, andere wählen den Validator dediziert für jeden Slot. Für den Leader Election Process ist eine Quelle, die Zufall produziert, notwendig [KRDO17, S. 1ff.].

### 3.3 Byzantine Fault Tolerance

Ein grundlegendes Problem in dezentralen, verteilten Systemen ist der Umgang mit dem Versagen einzelner Komponenten. Dies wird abstrakt dargestellt im *Problem der byzantinischen Generäle*, welches von Lamport, Shostak und Pease in *The Byzantine Generals Problem* [LaSP82] beschrieben ist.

Das Problem gibt ein Szenario vor in dem eine Gruppe von Generälen eine Stadt belagert. Jeder General ist in der Lage den anderen Nachrichten per Bote zu schicken. Ziel ist es sich gemeinsam auf einen Schlachtplan zu einigen. Zur Auswahl stehen die Aktionen *Angriff* und *Rückzug*.

Die Parallele zum Versagen einzelner Komponenten in verteilten Systemen wird gezogen, indem ein Anteil der Generärele verräterisch agiert und somit die Konsensbildung erschwert.

*Byzantine Fault Tolerance (BFT)* bezeichnet die Eigenschaft von Systemen, welche im beschriebenen Kontext fehlertolerant sind. Die Obergrenze an bösartigen Spielern bzw. fehlerhaften Komponenten beträgt dabei $\frac{1}{3}$.

Bezogen auf den Anwendungsfall hier stellen in dieser Analogie die einzelnen Validatoren die Komponenten dar, und das System bzw. das Verhalten des Gesamtsystems ist der Konsens bezüglich des nächsten Blocks.

## 4 Ansätze

Ansätze für Proof-of-Stake-Protokolle werden in chain-based PoS und BFT-style PoS unterschieden. Ersteres simuliert dabei das herkömmliche Mining, wohingegen letzteres einer Abstimmung gleicht.

### 4.1 Chain-based Proof of Stake

In chain-based Proof-of-Stake-Protokollen, auch *Blockchain Consensus* genannt, wird jedem Slot durch eine pseudozufällige Funktion ein Validator zugewiesen. Dies ist in Abbildung 2 zu erkennen [Zamf17, S. 1].

Ein Validator ist nun berechtigt einen Block zu stellen, der auf dem vorherigen Block aufbaut. Dies ist nicht immer eindeutig und bietet Potential für Nothing-at-Stake-Attacken (siehe dazu Abschnitt 5.4.2) [BuGr17, S. 1].

### 4.2 BFT-style Proof of Stake

Diese Familie von Proof-of-Stake-Protokollen basiert auf BFT-Algorithmen zur Konsensbildung. Validatoren haben alle das Recht ihre Stimme für vorgeschlagene Blöcke abzugeben. Dies ist in Abbildung 3 zu erkennen.

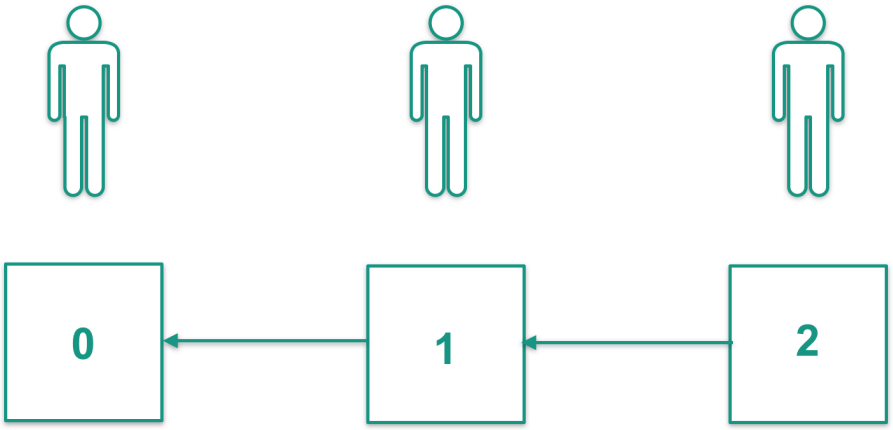


Abbildung 2: Chain-based Proof of Stake

Das Recht Blöcke vorzuschlagen wird dabei zufällig vergeben. Durch das Abstimmen ist es möglich auch schon ohne Metriken wie der *Longest Chain* Regel sicher zu sein, dass Konsens über Transaktionen bzw. über den Block in dem sie enthalten sind, herrscht[5][Naka08, S. 3]. Dies wird auch als *Finalization* bezeichnet [Zamf17, S. 1f.].

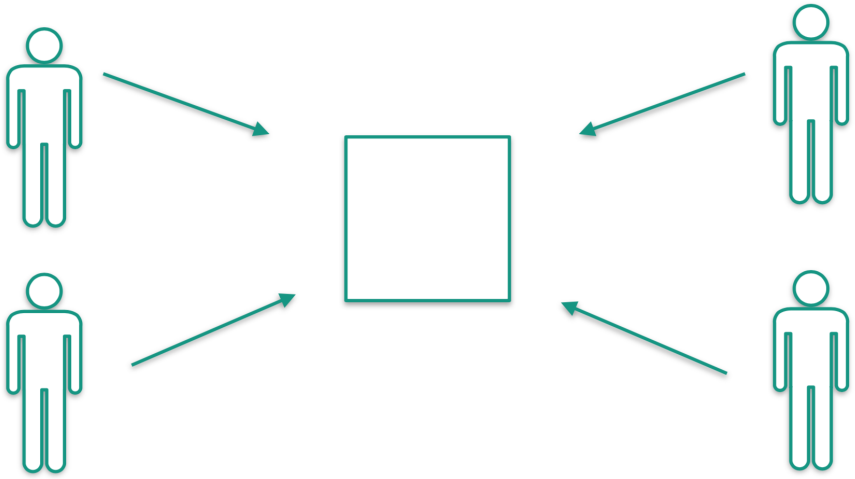

Abbildung 3: BFT-style Proof of Stake

Typisch für BFT-Algorithmen können bis zu einem Drittel fehlerhafte (oder bösartige) Teilnehmer toleriert werden.

### 4.3 Zufall

In PoW-Protokollen funktioniert der *Leader Election Process* ohne weiteres Zutun zufällig (siehe auch Abschnitt 1.2). Da dies in PoS-Protokollen fehlt, ist das Zuführen von Entropie notwendig [KRDO17, S. 1f.].

Dies kann auf verschiedene Weisen geschehen. Die einfachste Möglichkeit ist der Gebrauch eines externen Zufallsgenerators. Dies hat den Nachteil, dass eine externe Abhängigkeit geschaffen wird. Es entspricht auch nicht dem dezentralen Gedanken, welcher tief in der Philosophie hinter Kryptowährungen verankert ist.

[5]Vitalik Buterin und andere. Proof of Stake FAQ. `https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ`, aufgerufen am 28. Februar 2018

Durch Anwendung von *Secure Multi-Party Computation (MPC)* innerhalb des Protokolls, kann auch qualitativ hochwertiger Pseudozufall erzeugt werden. MPC beschreibt eine Familie von Protokollen, welche benutzt wird, um Funktionen mit mehreren geheimen Eingabeparametern verteilt zu berechnen. Dabei will keine der Parteien, welche die Parameter beisteuern, dass ihre geheimen Parameter bekannt werden. Nachteil hier ist jedoch, dass solche Berechnungen meist teuer sind und dementsprechend auch Zeit in Anspruch nehmen [Orla11, S. 1] [KRDO17, S. 3].

Eine andere Möglichkeit ist die Berechnung von Pseudozufall basierend auf angefallenen Daten, wie zum Beispiel Blockhashes etc., aus vorangegangen Blöcken. Ein Problem hierbei ist die Vorhersagbarkeit des Zufalls. Mehr dazu in den folgenden Abschnitten 5.1.1 und 5.1.2.

# 5 Evaluierung

Der Schwerpunkt der Evaluierung liegt, nach einer kurzen Einführung von Metriken, auf der Diskussion von Vorteilen und Nachteilen, sowie der Betrachtung einiger Angriffe. Soweit möglich im Vergleich zu PoW; bei neueren PoS-spezifischen Angriffen lediglich dediziert.

## 5.1 Merkmale von Proof-of-Stake-Protokollen

Da hochwertiger Zufall, z.B. durch MPC-Protokolle, teure Berechnungen oder externe Abhängigkeiten erfordert, wird oft auf Pseudozufall zurückgegriffen.

Abhängig von der Qualität des verwendeten Zufalls, macht das Heranziehen der Merkmale *Predictability* und *Recency* zur Analyse Sinn [eal18].

### 5.1.1 Predictability

Predictability wird unterteilt in *D-Local Predictability und D-Global Predictability*. Ein PoS-Protokoll ist *d-local predictable*, wenn es einem Validator möglich ist $d$ Blöck im Voraus zu wissen, ob er berechtigt ist, den betrachteten Block zu validieren.

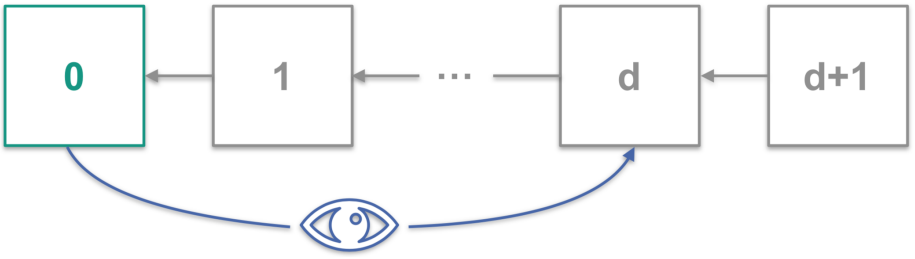


Abbildung 4: Predictability

Wenn es möglich ist für jeden beliebigen Validator herauszufinden, ob dieser in $d$ Blöcken berechtigt ist zu validieren, wird von *D-Global Predictability* gesprochen.

Jedes PoS-Protokoll ist *1-local predictable*, d.h. ein Validator kann, und soll, herausfinden, ob er berechtigt ist den nächsten Block zu validieren.

Bei PoS-Protokollen, welche Pseudozufall benutzen, um aus vorangegangen Informationen, wie z.B. Blockhashes, den Validator für künftige Blöcke zu bestimmen, kann die Predictibility helfen die Chance auf das Recht der Validierung künftiger Blöcke zu erhöhen. Erstellte Blöcke können überprüft und erst im Netzwerk propagiert werden, wenn sie hinsichtlich der Predictability optimal sind. Diese Eigenschaft kann auch bei Angriffen genutzt werden.

### 5.1.2 Recency

Das Merkmal Recency ist eine Negation der Predictability. *D-Recency* besagt somit, dass es für eine Partei nicht möglich ist $d$ Blöcke im Voraus herauszufinden, ob sie berechtigt ist den nächsten Block zu validieren.

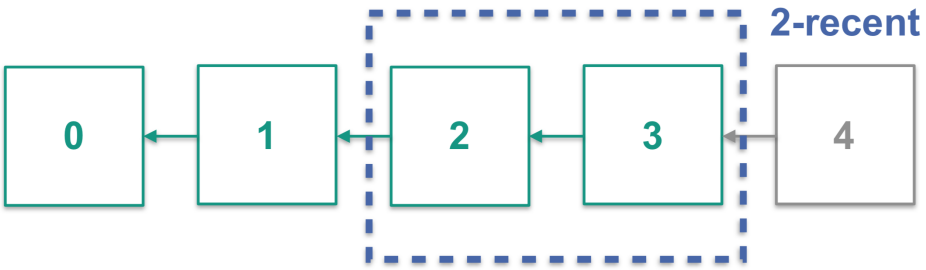


Abbildung 5: Recency

Die notwendigen Informationen zur Ermittlung der Berechtigung ergeben sich lediglich aus dem jüngsten Verlauf. Abbildung 5 zeigt die schematische Darstellung eines *2-recent* PoS-Protokolls.

## 5.2 Vor- und Nachteile von Proof-of-Stake-Protokollen

Im folgenden die Vor- und Nachteile von Proof-of-Stake-Protokollen. Nicht in jedem Fall sind diese klar ersichtlich. Oftmals sind es Trade-Offs wie z.B. bei der Gestaltung der Anreize. Ausschlaggebend ist in diesen Fällen die Präferenz des Protokolldesigners.

### 5.2.1 Energieverbrauch und Hardwarekosten

Wie eingangs erwähnt hatte alleine Bitcoin im Jahre 2017 einen Energieverbrauch von 29 TWh. Bei einem Strompreis von 29 ct./kWh[6] sind das 8,41 Mrd. Euro. Natürlich findet Mining nicht nur in Deutschland, oder Europa, statt. Daher müssen regional unterschiedliche Preise angesetzt werden, es soll hier nur die Größenordnung verdeutlicht werden.

Zusätzlich zu den Stromkosten, fallen auch Hardwarekosten an. Grafikkarten eigenen sich ebenfalls zum Minen und so stiegen mit der Beliebtheit der Kryptowährungen auch die Preise der Grafikkarten und es sank deren Verfügbarkeit.

Der Bedarf ist so groß, dass es sogar teure Eigenentwicklungen speziell zum Minen gibt, welche auf sog. ASICs (Application-Specific Integrated Circuits) basieren. Ein Beispiel dafür ist der *Antminer*[7] von Bitmain für über Zweitausend Dollar.

### 5.2.2 Bootstrapping Problem

Das Bootstrapping Problem im PoS-Kontext entsteht dadurch, dass das Staking, welches notwendig ist um die Transaktionen innerhalb der Kryptowährung als Validator zu bearbeiten, selbst diese Kryptowährung benötigt. Initial ist es also nicht ohne Weiteres möglich zu staken.

[6] Statistik entnommen aus `http://ec.europa.eu/eurostat/statistics-explained/index.php/Electricity_price_statistics/de`

[7] `https://shop.bitmain.com/antminer_s9_asic_bitcoin_miner.htm`

Ein Ausweg ist es die Münzen zu verkaufen. Dieser initiale Verkauf, der nicht nur PoS-Währungen vorbehalten ist, gleicht einem *Initial Public Offering (IPO)* und wird bei Kryptowährungen *Initial Coin Offering (ICO)* genannt.

Im Gegensatz zum PoW-Münzen bei denen alle Interessenten gleichberechtigt minen können, sind bei ICOs die Entwickler bzw. Verkäufer der Münzen im Vorteil.

Eine andere Möglichkeit ist das initiale Nutzen von PoW zur Ausschüttung der Münzen und ein späterer Umstieg auf PoS [BeGM16, S. 12f.].

### 5.2.3 Anreize

Die Gestaltung der Anreize in Bitcoin ist unterteilt in positive Anreize, wie dem Block Reward und den Transaktionsgebühren, sowie den negativen Anreizen in Form von Prozessorzeit und Elektrizitätskosten [Naka08, S. 4].

Diese negativen Anreize fallen jedoch immer an und sind nicht explizit mit bestimmten Aktionen verbunden. Minen, egal zu welchem Zweck, ist ein inhärent teurer Prozess.

Proof of Stake enthält die impliziten negativen Anreize des Minings nicht und kann bzw. muss diese daher explizit designen, um bei rational handelnden Spielern ein – für den Verlauf des Protokolls – wünschenswertes Verhalten zu generieren.

Negative Anreize sind immer mit dem Verlust von Teilen oder des ganzen Stakes verbunden. Dies wird im Falle der Kryptowährung *Ethereum*, im sich aktuell in Entwicklung befindlichen PoS-Protokoll *Casper*, Slashing genannt [BuGr17, S. 4f.].

Diese gewonnene Freiheit bei der Gestaltung der Anreize geht jedoch mit einer Steigerung der Komplexität einher.

### 5.2.4 Kontrolle über Ausgabe der Münzen

Eine Konsequenz der impliziten negativen Anreize in Proof-of-Work-Protokollen ist, dass auch die ehrliche Ausführung des Protokolls mit Kosten verbunden ist. Miner müssen, zusätzlich zum Profit den sie erzielen wollen, diese hohen Kosten decken. Dafür stehen Transaktionsgebühren und der Block Reward zur Verfügung. Diese müssen also zumindest so hoch sein um die anfallenden Kosten (Stromkosten, Hardwarekosten, etc.) zu decken. Das schränkt die freie Kontrolle über die Ausgabe der Münzen ein. Dies betrifft auch die negative Ausgabe, also zum Beispiel das Zerstören der gezahlten Transaktiongebühren für deflationäre Effekte.

Validatoren in Proof-of-Stake-Protokolle haben vernachlässigbare Strom und Hardwarekosten, dies gewährt PoS-Protokollen deutlich mehr Freiheit bei der Gestaltung der positiven und negativen Ausgabe.

### 5.2.5 Risiko der Zentralisierung

Das Risiko der Zentralisierung ist beim PoS-Protokollen geringer. Dies rührt daher, dass in Kryptowährungen, welche auf Proof of Work setzen, großes Kapital von Effekten der Massenproduktion profitieren kann. Die geringen Stückpreise führen in letzter Instanz zu überproportionaler Rechenleistung im Miningmarkt.[8]

[8] Vitalik Buterin und andere. Proof of Stake FAQ. `https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ`, aufgerufen am 28. Februar 2018

Zusätzlich zur Zentralisierung zum Kapital hin, findet auch eine Zentralisierung hin zu günstigen Stromkosten statt [Iv$^+$ot14, S. 4].

PoS-Systeme sind von diesen Risiken jedoch nicht betroffen. Validatoren werden nur anteilig an ihrem Stake bezahlt bzw. gewählt den Block zu validieren. Ihnen steht somit auch, je nach Design des Protokolls, nur anteilig zu Transaktionsgebühren und Block Reward zu gewinnen. Der Return of Invest ist also für alle Teilnehmer gleich.

### 5.2.6 Asymetrisches Verhältnis von Angriffs- und Verteidigungskosten

In einem Angriff wie in Abschnitt 2.4 beschrieben setzt in einem PoW-Kontext der Angreifer seine Rechenleistung ein, um – entgegen der Rechenleistung des Netzwerkes – eine längere Blockchain zur Verfügung zu stellen.

Die Kosten für einen Angreifer belaufen sich dabei auf die Stromkosten und ggf. noch die Opportunitätskosten wegen entgangener Block Rewards, falls der Angriff fehlschlägt.

Der Rechenleistung des Angreifers muss in gleichem Maße Rechenleistung auf der Seite der Verteidiger entgegengesetzt werden. Diese Symmetrie fehlt bei Angriffen auf Proof-of-Stake-Protokolle. Durch die Gestaltung der Anreize ist es möglich Strafen für unerwünschte Aktionen zu definieren, die zum Verlust von Teilen oder des komplettes Stakes des Angreifers führen.

Der Aufwand auf der Seite der Verteidiger liegt nur darin den Angriff zu bemerken und die Mechaniken des Protokolls in Gang zu setzen, die zum Verlust des Stakes beim Anreifer führen (z.B. *Evidence Transactions* bei Tendermint). Es kann hier also eine Asymmetrie festgestellt werden[9].

### 5.2.7 Ansprüche an die IT-Sicherheit

Angriffe auf die Systeme eines Miners oder eines Validators, d.h. insbesondere die Fremdsteuerung, um z.B. einen Fork zu erstellen, sind durch die negativen Anreize mit Kosten verbunden. Diese wurden im vorangegangenen Abschnitt besprochen.

Aus dem asymetrischen Verhältnis, welches vorteilhaft ist, um Anreize für gewünschte Verhaltensweisen zu bieten, ergibt sich in einem Proof-of-Stake-Kontext auch ein höheres Risiko im Falle eines unverschuldeten Fehlverhaltens. Daher sollte dieses Risiko dementsprechend durch Versicherungen oder Investitionen in die IT-Sicherheit bedacht werden.

## 5.3 Angriffe

Es gibt diverse Angriffe auf PoS-Protokolle, einige wurden dabei von den PoW-Protokollen geerbt, es gibt jedoch auch spezifische Angriffe auf Proof of Stake.

### 5.3.1 Predictable Selfish Mining

Selfish Mining existierte schon zuvor als Angriff auf PoW-Protokolle. Der Vorteil gegenüber den ehrlichen Minern wird beim Selfish Mining daraus bezogen, dass die ehrlichen Miner gezwungen werden Rechenzeit auf einem überholten öffentlichen Branch zu verschwenden.

[9] Vitalik Buterin. A Proof of Stake Design Philosophy. `https://medium.com/@VitalikButerin/a-proof-of-stake-design-philosophy-506585978d51`, aufgerufen am 28. Februar 2018

Dies geschieht indem ein Miner, welcher einen Block gefunden hat, die Propagation des Blocks im Netz hinauszögert. Die anderen Miner werden dadurch gezwungen weiterhin auf dem alten Block zu minen.

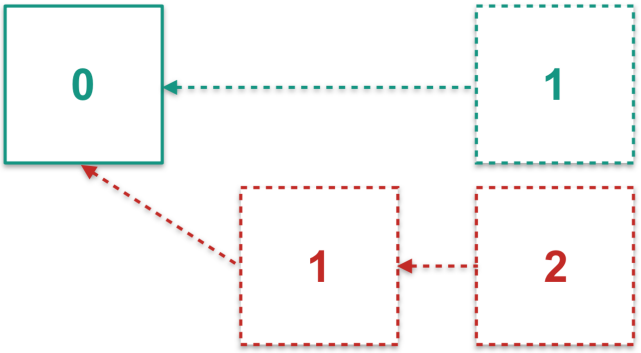


Abbildung 6: Selfish Mining Angriff

Der so erlangte Vorsprung kann genutzt werden, um nun nach einem neuen Block zu suchen. Der Angriff ist für den Selfish Miner mit dem Risiko behaftet selbst Rechenzeit zu verschwenden, falls ein Block gefunden wird, bevor er seinen vorgehaltenen Block veröffentlicht [EySi14, S. 440ff.] [CoBa14, S. 6ff.].

Durch Local Predictability kann das eigene Risiko minimiert werden, indem die Blöcke so gestaltet werden, damit die Chance auf die Berechtigung den nächsten Block validieren zu können steigt (siehe auch *Stake Grinding* in Abschnitt 5.4.1).

Protokolle mit hoher Global Predictability erlauben eine noch bessere Abschätzung des Risikos, indem Informationen über andere Parteien ebenfalls in die Betrachtung des Risikos einbezogen werden können. Durch Angriffe auf diese Ziele kann dieses Risiko sogar beeinflusst werden.

## 5.4 Predictable Double Spending

Double Spending ist ebenfalls schon als Angriff auf Bitcoin bekannt und tatsächlich sogar der Grund wieso Bitcoin bzw. Proof of Work in diesem Kontext überhaupt existiert. Vor Bitcoin gab es keine Möglichkeit dieses Problem ohne das Hinzuziehen vertrauenswürdiger Drittparteien zu lösen.

Beim Double Spending geht es darum gültige Transaktionen umzukehren, indem ein Branch veröffentlicht wird, welcher eine Transaktion enthält, die im Konflikt steht [KRDO17, S. 45].

Abbildung 7 verdeutlicht das. Zwischen Block 0 und 1 wird eine Transaktion getätigt, welche Währungseinheiten, im Tausch für Ware oder andere Kryptowährungen, verkauft. Diese Transaktion ist im grünen Block 1 enthalten. Wenn die Ware erhalten ist, wird der längerer rote Fork veröffentlicht, welcher eine Transaktion enthält, die im Konflikt steht.

Es ist nicht notwendig für den Angreifer den längeren Branch selbst zu minen. In sog. *Bribe Attacks* wird anderen Minern eine Bestechung gezahlt um den eigenen Branch zu minen. Dies kann helfen den Angriff effizienter zu gestalten, wenn die Kosten für die Bestechung geringer sind als die Stromkosten, die beim alleinigen Minen entstehen würden [BeGM16, S. 3].

Im Proof-of-Stake-Kontext sind die Kosten für einen Angriff geringer, wenn es keine weiteren Mechanismen zur Verhinderung gibt. Es können sich also auch Double-Spending-Angriffe über länger Zeiträume hinweg, *Longe-Range Revision Attack* genannt, lohnen. Dort ist es

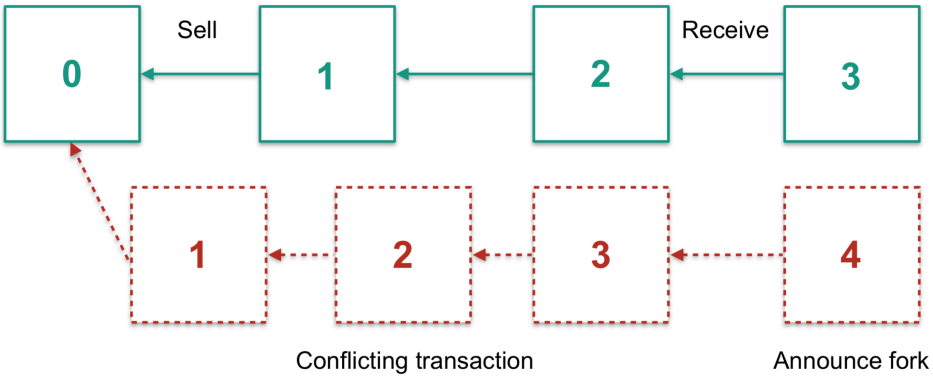


Abbildung 7: Double Spending Angriff

nicht einmal notwendig, dass der Angreifer den Stake zum aktuellen Zeitpunkt überhaupt noch hält.

Angriffe die darauf abzielen einen anderen Branch zu veröffentlichen, können durch das Konzept der Finalization verhindert werden [BuGr17, S. 5].

### 5.4.1 Stake Grinding

In Stake-Grinding-Angriffen wird versucht den Zufall, durch welchen die Berechtigung einen Block zu validieren bestimmt wird, zu beeinflussen – kurz: den Leader Election Process. Es werden verschiedene Blockheader getestet, um so die Chance zu erhöhen.

Dieser Vorgang gleicht in gewisser Weise dem Mining bei Proof-of-Work-Protokollen. Diese sind nicht anfällig für Stake-Grinding-Angriffe, da dort der Zufall nicht beeinflusst werden kann. Proof-of-Stake-Protokolle, welche externen Zufall benutzen oder Zufall durch MPC-Protokolle generieren, sind ebenfals nicht anfällig für diese Art von Angriffen [KRDO17, S. 46].

### 5.4.2 Nothing at Stake

Nothing at Stake beschreibt eine Situation in welcher rationale Miner oder Validatoren mehrere Branches vorfinden und analysieren welchen sie – oder sogar, ob sie beide – minen bzw. validieren sollen.

*Nothing at Stake im Proof-of-Work-Kontext* — Die Bezeichnung Nothing at Stake ist bei Proof of Work etwas irreführend, da zu jedem Zeitpunkt die Rechenzeit (bzw. der ökonomische Wert der Rechenzeit) „at Stake“, also eingesetzt, ist.

Abbildung 8 zeigt dieses Szenario. Nach einem Fork sind zwei Branches zu sehen. Der linke Branch ist der Branch mit dem höheren Erwartungswert.

Ein Miner hat nun vier Möglichkeiten:

1. **Keinen Branch minen**
   In dem Fall ist der Erwartungswert 0. Durch Inaktivität wird nicht am Leader Election Process teilgenommen.

[10] Quelle der Abbildung: Vitalik Buterin und andere. Proof of Stake FAQ. `https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ`, aufgerufen am 28. Februar 2018

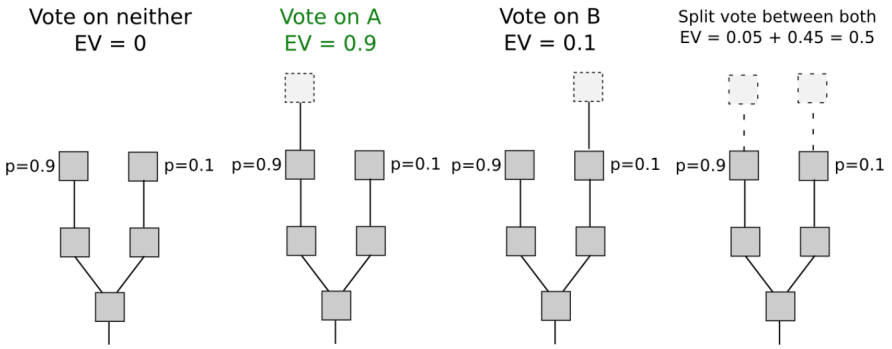


Abbildung 8: Nothing at Stake in einem Proof-of-Work-Kontext[10]

2. **Linken Branch minen**
   Die Wahrscheinlichkeit den Leader Election Process zu gewinnen ist in beiden Branches gleich hoch. Für den linken Branch ist jedoch die Wahrscheinlichkeit höher Teil der kanonischen Blochchain zu sein. In diesem Fall ist die Arbeit, die in das Minen des linken Branches fließt, nicht verschwendet.

3. **Rechten Branch minen**
   Der rechte Branch hat eine geringere Chance Teil der kanonischen Blockchain zu sein. Arbeit, die hier in das Minen geht, ist somit weniger rentabel. Ein rationaler Miner wird also den linken Branch bevorzugen.

4. **Beide Branches minen**
   Beim Minen von beiden Branches teilt sich die verfügbare Rechenkraft auf zwei Branches auf. Somit entsteht ein Nachteil gegenüber den Minern auf der kanonischen Blockchain.

Es lässt sich erkennen, dass außerhalb des Protokolles – durch die entstehenden Kosten in der realen Welt – negative Anreize gegeben sind. Diese führen bei rationalen Minern, ohne weitere Maßnahmen dazu, die kanonische Blockchain zu minen bzw. den Branch bei dem die Zugehörigkeit zur dieser am höchsten eingeschätzt wird.

*Nothing at Stake im Proof-of-Stake-Kontext* — Ohne explizite Maßnahmen dagegen erscheint Nothing at Stake im Proof-of-Stake-Kontext deutlich attraktiver, da die negativen Anreize, welchen durch das Mining gegeben sind, fehlen.

Auch hier hat ein rationaler Validator vier Möglichkeiten zu reagieren:

1. **Keinen Branch minen**
   Wie im PoW-Bespiel macht es auch hier ökonomisch keinen Sinn sich zu enthalten.

2. **Linken Branch minen**
   Der linke Branch hat auch hier eine höhere Chance teil der kanonischen Blockchain zu sein, somit ist der Erwartungswert höher.

3. **Rechten Branch minen**
   Der rechte Branch hat eine niedrigere Chance teil der kanonischen Blockchain zu sein.

4. **Beide Branches minen**
   Da keine Rechenkraft aufgewandt werden muss, und bei Abwesenheit weiterer negativen Anreize, kann ein rationaler Validator den Erwartungswert erhöhen, wenn er in beiden Branches validiert. Somit geht das Risiko einen nicht-kanonischen Branch zu validieren auf Null.

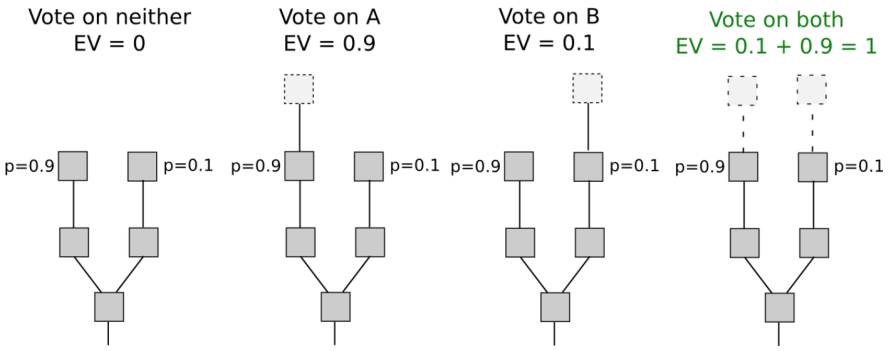


Abbildung 9: Nothing at Stake in einem Proof-of-Stake-Kontext[11]

Im Proof-of-Stake-Kontext wird sich nur ein altruistischer Validator verhalten wie im Falle von Proof of Work. Auszugehen ist jedoch von rationalen Validatoren, die geleitet von ihrem ökonischen Interesse ihren Gewinn maximieren wollen. Daher müssen diese sogar beide Branches validieren, wenn sie sich rational verhalten wollen [KRDO17, S. 47f.].

Um diese Nothing-at-Stake-Situation zu entschärfen, können negative Anreize eingeführt werden, z.B. das *Slashing* im Falle von *Ethereums* PoS-Protokoll *Casper* [BuGr17, S.1, S. 5f.].

### 5.4.3 Catastrophic Crashes

BFT-style Proof-of-Stake-Algorithmen benötigten 2/3 Mehrheiten für die Protokollausführung (z. B. für die Finalization). Sie sind daher besonders anfällig für Wegfall von Validatoren aus dem Netzwerk. Dies wird Catastrophic Crash genannt. Der Wegfall muss nicht unbedingt durch einen Angriff erfolgen, er kann auch durch Begebenheiten im Netz entstehen.

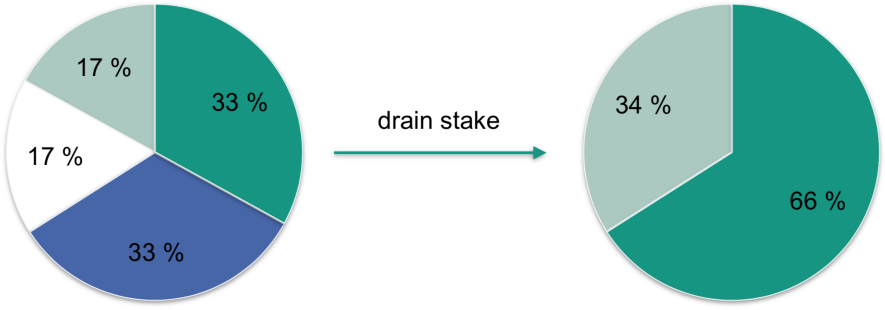


Abbildung 10: Zu sehen ist die initiale Konfiguration der Stakes der Validatoren mit anschließendem Stake Draining.

Abbildung 10 zeigt links eine Konfiguration von Validatoren anhand ihres Stakes. Es gibt vier Validatoren, zwei mit jeweils 17 % Anteil (weiß und helles Grün) und zwei mit 33 % Anteil (blau und dunkles Grün) am gesamten Stake. Nach Wegfall des weißen und blauen Validators können die beiden grünen Validatoren nur noch 50 % der Stimmkraft aufbringen. Das BFT-style PoS-Protokoll kann so nun nicht mehr fortfahren.

Durch Verringern der Stakes der inaktiven Validatoren, auch *Stake Draining* genannt, ist es möglich wieder zu einer Konfiguration der Stakes zu kommen, in welcher wieder Mehrheiten erreicht werden können. Dies ist im rechten Teil von Abbildung 10 zu sehen [BuGr17, S. 8f.].

[11] Siehe Fußnote 10

# 6 Ausblick

Der größte Vorteil von Proof-of-Work-Protokollen ist ihre Einfachheit. Durch das Mining sind negative Anreize implizit vorhanden und müssen nicht explizit ausgestaltet werden.

Mit steigendem Verständnis für Kryptowährungen kann diese Einfachheit jedoch einen Nachteil darstellen. Die Möglichkeit Anreize selbst frei zu gestalten ist verwehrt und somit kann das Verhalten der Partizipanten nicht nach Belieben beeinflusst werden.

Das Proof-of-Work-Verfahren für Kryptowährungen ist trotzdem nicht obsolet. Es eignet sich als gute Lösung für die initiale Verteilung der Münzen – das Bootstrapping Problem. Und so ist eine Koexistenz dieser beiden Methoden vorstellbar und sogar wünschenswert.

Interessante Kryptowährungen im Kontext von Proof of Stake sind momentan Ethereum[12] und Cardano[13]. Beide sind beliebte Top 10[14] Münzen mit einer hohen Marktkapitalisierung. Ethereum setzt aktuell auf Proof of Work und plant den graduellen Umstieg auf Proof of Stake[15]. Das Problem des Bootstrappings wurde hier also umgangen. Cardano setzt auf ein reines Proof-of-Stake-Verfahren und nutzte einen ICO um die Münzen initial zu verteilen. Es ist die Einführung von Stake Delegation geplant[16]. Dies gibt kleinen Usern die Möglichkeit sich zu Pools zusammenzuschließen und – parallel zu Miningpools – gemeinsam zu validierern.

Die Adaption durch größere Projekte wird Proof of Stake mehr Gewicht geben und das Interesse und die Forschung in dem Bereich vorantreiben.

[12] Siehe `https://www.ethereum.org/`

[13] Siehe `https://www.cardanohub.org/`

[14] Daten zur Markkaptitalisierung sind entnommen aus `https://coinmarketcap.com/`, aufgerufen am 1. März 2018

[15] Siehe auch `https://cryptocanucks.com/metropolis-part-2-constantinople-ethereum/`

[16] Stake Delegation wird mit dem Shelley Release erscheinen, siehe `https://cardanoroadmap.com/#shelley-decentralised`

## Literatur


[Ba+ot02] Adam Back und andere. Hashcash - A Denial of Service Counter-Measure, 2002.

[BeGM16] Iddo Bentov, Ariel Gabizon und Alex Mizrahi. Cryptocurrencies without Proof of Work. In *International Conference on Financial Cryptography and Data Security*. Springer, 2016, S. 142–157.

[BuGr17] Vitalik Buterin und Virgil Griffith. Casper the Friendly Finality Gadget. *CoRR* Band abs/1710.09437, 2017.

[CoBa14] Nicolas T. Courtois und Lear Bahack. On Subversive Miner Strategies and Block Withholding Attack in Bitcoin Digital Currency. *CoRR* Band abs/1402.1718, 2014.

[eal18] Jonah Brown-Cohen et al. Formal Barriers to Proof-of-Stake Protocols. Blockchain Protocol Analysis and Security Engineering 2018, 2018.

[EySi14] Ittay Eyal und Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security*. Springer, 2014, S. 436–454.

[Iv+ot14] Come-from-Beyond [Sergey Ivancheglo] und andere. Nxt Whitepaper. `https://bravenewcoin.com/assets/Whitepapers/NxtWhitepaper-v122-rev4.pdf`, 2014.

[KRDO17] Aggelos Kiayias, Alexander Russell, Bernardo David und Roman Oliynykov. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. 2017.

[LaSP82] Leslie Lamport, Robert Shostak und Marshall Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)* 4(3), 1982, S. 382–401.

[Naka08] Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. `http://bitcoin.org/bitcoin.pdf`, 2008.

[Orla11] Claudio Orlandi. Is Multiparty Computation Any Good In Practice? In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, S. 5848–5851.

[Zamf17] Vlad Zamfir. Casper the Friendly Ghost A 'Correct-by-Construction' Blockchain Consensus Protocol. `https://github.com/ethereum/research/tree/master/papers/CasperTFG`, 2017.