Mining und Konsens in Bitcoin

Nico Daßler

Friedrich-Alexander Universität Erlangen-Nürnberg,
nico.dassler@fau.de

Zusammenfassung. In konventionellen Währungssystemen sorgt eine zentrale Instanz für die Validierung jeder einzelnen Transaktion. Es wird überprüft, ob der Absender die Geldmittel tatsächlich besitzt und ob er autorisiert ist, auf diese zuzugreifen. Für digitale Währungen werden ähnliche Überprüfungen durchgeführt, allerdings im Falle von Bitcoin dezentralisiert, um das System vor Manipulation zu schützen. Damit in einem solchen System die Sicherheit gewährleistet ist, wird das sogenannte *Proof-of-Work*-Verfahren eingesetzt. In diesem Verfahren werden im Wettbewerb zwischen den Minern Blöcke für die Blockchain erzeugt, welche die Transaktionen zusammenfassen und aufzeichnen. Dabei wird vom Netzwerk immer die Kette von Blöcken bevorzugt, welche am längsten ist. Das bedeutet, dass durch das *Proof-of-Work*-Verfahren die Kette ausgewählt wird, in die der größte Aufwand investiert wurde. Auf diese Weise wird die Richtigkeit der Blockchain sichergestellt solange der Großteil der Rechenleistung immer auf der ehrlichen Seite liegt. Um das sicherzustellen, wird für jeden erfolgreich erzeugten Block eine Belohnung an den Miner in Form von Bitcoin ausgezahlt.

1 Einführung

Jede Art von Währung muss dem Betrug mit dieser Währung entgegenwirken. Bei gedrucktem Geld zum Beispiel muss verhindert werden, dass Falschgeld hergestellt werden kann. Dies wird durch spezielle Sicherheitsmerkmale auf Geldscheinen erreicht, welche den Nachdruck unmöglich machen sollen. Bei digitalen Währungen gibt es ähnliche Probleme. Zum Beispiel muss verifiziert werden, dass bei einer Transaktion kein kopiertes Geld zum Einsatz kommt. Dazu gehört das sogenannte *Double-Spend-Problem*, bei dem ein bestimmter Betrag mehrfach ausgegeben wird. In konventionellen Währungssystemen tritt dieses Problem beim Austausch von Bargeld nicht auf, da der Wert an den physischen Geldschein gebunden ist, der nicht geteilt werden kann. Bei (Online-) Überweisungen allerdings muss durch eine zentrale Instanz jede Transaktion validiert werden. Eine solche zentrale Instanz ist aber anfällig für Manipulationen und Betrug (*Single Point of Failure*). Bitcoin nutzt eine verteilte Architektur, um dieses Problem zu umgehen, welche allerdings die Behandlung des *Double-Spend-Problems* erschwert, da über die verschiedenen Netzwerkknoten Konsens über bereits durchgeführte Transaktionen hergestellt werden muss.

In der folgenden Arbeit wird beschrieben, wie Bitcoin das Konsensproblem mithilfe von Mining löst. Zunächst werden hierfür einige Grundlagen von Bitcoin

erläutert, gefolgt von einer Beschreibung der Informationsverbreitung in Bitcoin. Danach wird das Mining erläutert, mit dem Transaktionen aufgezeichnet werden und es wird beschrieben, wie die verschiedenen Systeme zusammenarbeiten, um Konsens und Sicherheit in Bitcoin herzustellen. Zuletzt wird die Funktionsweise des Systems zusammengefasst und bewertet.

2 Grundlagen

2.1 Transaktionen

Um mit Bitcoin Transaktionen durchzuführen, wird mithilfe eines sogenannten Wallets (dt. Geldbeutel) eine Transaktion durchgeführt. Die Wallet-Software speichert die privaten und öffentlichen Schlüssel, die zur Signatur von Transaktionen und zur Verifikation des Besitzes der Bitcoins eingesetzt werden. Außerdem bildet die Wallet-Software bei Bedarf eine syntaktisch korrekte Transaktion und speist diese in das Bitcoin-Netzwerk ein. Die für diese Arbeit wichtigen Datenfelder und Mechanismen einer Transaktion werden in den beiden folgenden Abschnitten beschrieben.

2.2 Signaturen in Bitcoin

Digitale Signaturen werden eingesetzt, um zu beweisen, dass ein beliebiger Datenstrom (zum Beispiel eine E-Mail) tatsächlich von einer bestimmten Person verfasst wurde. Dabei wird mithilfe des Datenstroms und des privaten Schlüssels eine Signatur berechnet, welche dann mithilfe des öffentlichen Schlüssels verifiziert werden kann. Mit dem Datenstrom, dem öffentlichen Schlüssel und der Signatur kann jeder sicherstellen, dass der Datenstrom vom Besitzer des privaten Schlüssels verfasst wurde. In Bitcoin wird der *Elliptic Curve Digital Signature Algorithm* eingesetzt, um den Besitz von Bitcoin zu beweisen [2].

In Bitcoin ist die Nachricht, die signiert wird, die Transaktion. Dabei können Teile oder die gesamte Transaktion in der Signatur verwendet werden. Das ist in verschiedenen Szenarien nützlich [2]. Mithilfe von Signaturen und deren Verifikation werden unteilbare Geldmittel (engl. *funds*) *unlocked* beziehungsweise für die Nutzung in einer weiteren Transaktion freigeschaltet. Diese Geldmittel werden *UTXOs* genannt. UTXO steht für einen *unspent transaction output* [2], also ein Geldmittel, dass noch nicht ausgegeben wurde. Das Freischalten und Sperren wird in Form von Skripten innerhalb einer Transaktion spezifiziert.

2.3 Aufbau und Funktionsweise von Transaktionen

Abbildung 1 zeigt den Aufbau einer Transaktion. Das Feld *vout* beschreibt die neuen UTXOs (eine Liste), die bei der Transaktion angelegt werden. Jeder dieser UTXOs hat einen Betrag und ein Skript, das zur Freischaltung des Betrags erfüllt werden muss. Das Skript ist in der Sprache *Script* geschrieben. Mit einem solchen Skript werden Signaturen erzeugt und geprüft. In den meisten Fällen wird der

Version und locktime	---	
vin	txid und vout	Diese beiden Felder referenzieren, die UTXOs die bei der Transaktion überwiesen werden.
	scriptSig	Dieses Skript dient zum freischalten des referenzierten UTXOs.
vout	value	Der Betrag der UTXOs, die bei bei der Transaktion erstellt werden, in Satoshis
	script Pubkey	Dieses Skript sperrt den Betrag.

Abb. 1. Zeigt die für diese Arbeit relevanten Felder einer Transaktion.

UTXO mit dem *Pay-to-Public-Key-Hash* [2] Skript gesperrt. Das bedeutet, dass zum Entsperren eine Signatur über die Transaktion benötigt wird. Im Feld *vin* werden die Geldmittel spezifziert, die versendet werden. Dies passiert über die Referenz auf UTXOs, die in *vout* Abschnitten von anderen Transaktionen stehen.

Auf diese Weise entsteht eine Verkettung von Transaktionen. Die UTXOs einer Transaktion dienen als Geldmittel für eine andere Transaktion, welche wieder neue UTXOs anlegt und wieder als Geldmittel verwendet werden können. Die UTXOs sind dabei unteilbar. Soll zum Beispiel eine Transaktion über 5 Bitcoin durchgeführt werden und es steht nur ein UTXO in Höhe von 10 Bitcoin zur Verfügung[1], so muss der Sender zwei UTXOs im vout Feld definieren. Einen für den Empfänger über 5 Bitcoin und einen für sich selbst als Wechselgeld.

Bei jeder Transaktion werden außerdem Gebühren durch den Sender definiert. Diese Gebühren sind nicht notwendig, allerdings beschleunigen sie die Verifikation der Transaktion und spielen eine wichtige Rolle beim Mining (Abschnitt 4). Die Gebühren werden über die Differenz der vin und der vout Beträge implizit angegeben.

2.4 Block

Die im letzten Abschnitt beschriebenen Transaktionen werden in sogenannten Blöcken aggregiert, um innerhalb des Bitcoin-Netzwerks eine einheitliche Sicht (Konsens) darüber zu schaffen, welche UTXOs existieren und welche bereits ausgegeben wurden. Ein Block beginnt mit einigen Metadaten und dem *Block Header*. Der Block Header enthält einige für das Mining und der Konsensfindung relevante Felder, welche im Folgenden erklärt werden.

Abbildung 2 zeigt den Aufbau des Block Headers. Der Hash vom letzten Block (beziehungsweise vom letzten Block Header) ist eines der wichtigsten Elemente im Block. Er sorgt dafür, dass Blöcke miteinander verknüpft sind. Die

[1] Der Betrag eines UTXOs wird nur einmal im vout Feld einer Transaktion spezifiziert. Im vin ist der Betrag implizit über die Referenz auf den UTXO enthalten.

Version	Dient zum Festhalten von Protokoll- oder Software-Upgrades.
Hash vom letzten Block	Hash vom letzten Block Header. Dient zur Verkettung der Blöcke.
Merkle Root	Spezieller Hash der im Block enthaltenen Transaktion. Die Transaktionen werden so im Header referenziert.
Zeitstempel	Gibt den ungefähren Zeitpunkt der Erstellung des Blocks an.
Schwierig-keitsgrad	Gibt an wie schwierig es ist/war den Block zu erzeugen (minen).
Nonce	Hier können 32-Bit zufällige Daten eingetragen werden.

Abb. 2. Zeigt den Aufbau des Block Headers.

Blöcke formen damit eine Kette von Blöcken, welche als Blockchain bezeichnet wird. Die Merkle Root ist eine Zusammenfassung der im Block enthaltenen Transaktionen [2]. Sie stellt sicher, dass eine Veränderung der Transaktionen eine Veränderung des Headers zur Folge hat. Der Schwierigkeitsgrad und die Nonce werden bei der Erstellung (Mining) eines Blocks verwendet. Um einen gültigen Block zu erzeugen, muss der Hash des Headers kleiner sein als der Schwierigkeitsgrad (auch *Difficulty Target* genannt), der im Header spezifiziert wird. Da das Ergebnis einer Hashfunktion nicht vorhergesagt werden kann, müssen verschiedene Header ausprobiert werden. Das Nonce Feld im Header dient zur Veränderung des Headers und damit zur Veränderung des Hashwerts. Andere Felder dürfen nicht verändert werden, da der Header sonst nicht mehr gültig ist. Um das Target zu unterschreiten, muss Rechenleistung aufgebracht werden (testen vieler verschiedener Nonce Werte). Dass diese Leistung tatsächlich erbracht wurde, kann mit einem einzelnen Durchlauf durch die Hashfunktion und anschließendem Vergleich mit dem *Difficulty Target* verifiziert werden. Dieser Vorgang wird *Proof-of-Work* genannt.

2.5 Betrug in Bitcoin

Mit den in Abschnitt 2.1 besprochenen Mechanismen kann ein Empfänger von Bitcoins bestätigen, dass der Absender der tatsächliche Besitzer der Bitcoins ist. Der Empfänger kann die Signaturen verifizieren, um die Kette des Besitzes zurückzuverfolgen [4]. Es ist also nicht möglich Bitcoins zu kopieren oder zu erzeugen, da die Bitcoins bis zu ihrem Ursprung zurückverfolgt werden können. Die Transaktionen werden dafür in der Blockchain gespeichert (siehe Abschnitt 2.4). Dabei stellt sich die Frage, wie in einem verteilten System Einigkeit über den Zustand der Blockchain entstehen kann und wie diese gegen Veränderung geschützt wird. Wäre es möglich eine Transaktion im Nachhinein zu verändern, so könnte ein Käufer dieselben Bitcoins mehrfach ausgeben. In den nachfolgenden Abschnitten wird die Lösung dieses sogenannten *Double-Spend-Problems* durch Mining und das Befolgen einiger Regeln durch jeden Netzwerkknoten beschrieben.

3 Informationsverbreitung

In den folgenden beiden Abschnitten werden die Regeln beschrieben, mit denen Transaktionen und Blöcke vom Bitcoin-Netzwerk akzeptiert und verteilt werden. Das Netzwerk ist ein *Peer-to-Peer Netzwerk*. Die Knoten in einem P2P Netzwerk sind gleichgestellt, jedoch nehmen sie im Bitcoin-Protokoll unterschiedliche Aufgaben wahr. Nachfolgend werden Knoten, die Routing Aufgaben übernehmen und eine komplette Kopie der Blockchain enthalten als *Full Nodes* bezeichnet. Als Wallet werden Knoten bezeichnet, die lediglich die Schlüssel eines Nutzers verwalten und sich an Full Nodes wenden, um Zugriff auf die Blockchain zu erhalten. In Abschnitt 4 wird zusätzlich der Begriff *Miner* für Knoten verwendet, die Mining durchführen. Abbildung 3 zeigt einen Teil dieser Komponenten in einem einfachen Bitcoin-Netzwerk.

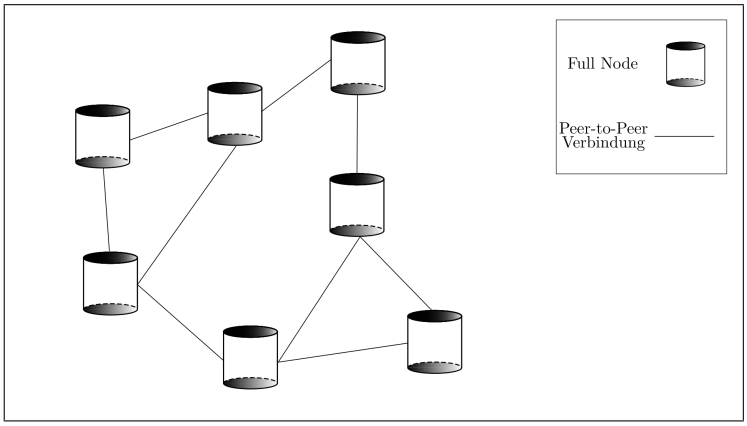

Abb. 3. Der Aufbau eines einfachen Bitcoin-Netzwerks mit Peer-to-Peer Verbindungen und Full Nodes.

3.1 Verbreitung von Transaktionen

Eine Transaktion wird typischerweise in einem Wallet-Knoten erzeugt. Zum Beispiel wenn ein Nutzer einen anderen für einen Service oder ein Produkt bezahlen möchte. Die Transaktion wird entsprechend der in Abschnitt 2.1 beschriebenen Struktur aufgebaut und anschließend über eine Full Node an das P2P Netzwerk übergeben. Beim Erhalt einer Transaktion werden dessen Struktur und Inhalt überprüft. Falls alle Kriterien erfüllt sind, wird die Transaktion an benachbarte Knoten weitergegeben und in den lokalen Speicher für unbestätigte Transaktionen übertragen (auch *Memory Pool* oder *Transaction Pool* genannt). In Abbildung 4 wird dieser Vorgang mit einer einzelnen Transaktion dargestellt. Die Transaktion wird in vereinfachter Form nur über die ID (txid) dargestellt, die normalerweise 256-Bit lang ist.

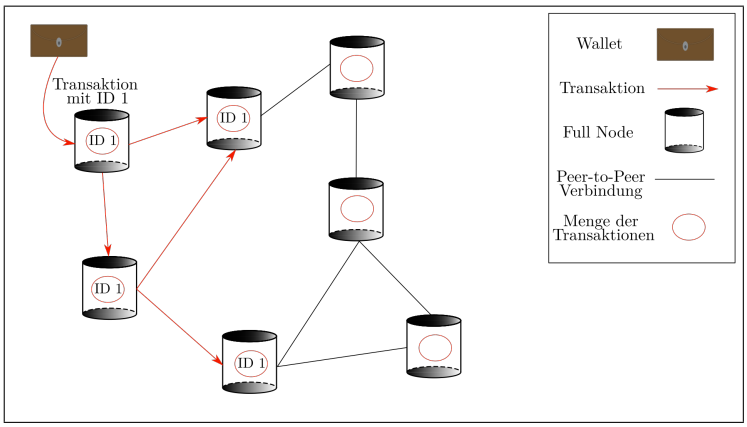

Abb. 4. Die Verbreitung einer Transaktion im P2P Netzwerk. Nach der Überprüfung einer Transaktion wird diese im Memory Pool gespeichert und an benachbarte Knoten weitergeleitet.

Um zu bestätigen, dass eine Transaktion gültig ist, wird zunächst deren Aufbau auf Korrektheit überprüft. Danach werden einige formale Aspekte des Inhalts überprüft wie zum Beispiel ob die Maximallänge überschritten wurde oder die vin- und vout-Liste beide mindestens ein Element haben. Weitere wichtige Kriterien[2] sind:

- Die txids von allen vin Elementen müssen im Transaction Pool oder in einem Block der Blockchain[3] enthalten sein. Falls keine passende Transaktion für eines der Elemente gefunden wird, wird die Transaktion dem *Orphan Transaction Pool* hinzugefügt.
- Keiner der Inputs (vin Elemente) darf in einer anderen Transaktion als Input vorkommen, da er sonst bereits ausgegeben wurde. Das gilt auch für Transaktionen mit dem gleichen Input im Orphan Transaction Pool.
- Die Skripte zum Sperren und Entsperren für jedes Input-Output-Paar müssen ein gültiges Ergebnis liefern.

Bei Verletzung einer der oben genannten Regeln wird diese Transaktion verworfen und nicht im Netzwerk verbreitet. Alle ehrlichen Full Nodes verhindern somit die Verbreitung von falschen oder betrügerischen Transaktionen. Es werden zum Beispiel Transaktionen aussortiert, deren Ersteller nicht Besitzer der UTXOs ist oder wenn eine Transaktion versucht, einen UTXO zweimal auszugeben. Durch diese einfachen Regeln entsteht damit Konsens im Netzwerk, welche Transaktionen gültig sind und welche nicht. Allerdings können doppelt ausgegebene UTXOs

[2] Es gibt weitere Kriterien, die allerdings weniger relevant für das Verständnis des Minings und der Konsensfindung sind. Diese Regeln sind in [2] oder im Bitcoin Source Code [1] aufgelistet.

[3] Für schnelleren Zugriff werden diese im UTXO Set abgespeichert.

nur mithilfe der Blockchain aussortiert werden. Im nächsten Abschnitt wird eine weitere Menge von Regeln beschrieben, die im Netzwerk Konsens über den Zustand der Blockchain herstellen.

3.2 Verbreitung von Blöcken und Aufbauen der Blockchains

Nachdem ein Block von einem Mining Knoten erstellt wurde (siehe Abschnitt 4), wird dieser an das Netzwerk übergeben. Ähnlich wie bei der Verbreitung von Transaktionen wird eine Menge von Regeln für jeden Block überprüft und im Falle einer Regelverletzung verworfen. Der Block muss die in Abschnitt 2.4 beschriebene Struktur aufweisen. Außerdem muss der Hash des Headers kleiner sein als das angegebene *Difficulty Target* und alle enthaltenen Transaktionen müssen die in Abschnitt 3.1 aufgeführten Regeln einhalten. Durch diese Regeln wird sichergestellt, dass keiner der Miner betrügt. Versucht ein Miner zum Beispiel falsche Transaktionen in einen Block einzubauen, wird der Block vom Netzwerk abgelehnt und nicht weiter verbreitet.

Wenn eine Full Node einen Block erhält, der die Regeln einhält, wird er versuchen diesen an die Blockchain anzuhängen. Dafür muss der Knoten herausfinden, welcher Block in der Blockchain der Vorgänger des neuen Blocks ist. Das Headerfeld Hash vom letzten Block wird dafür mit den letzten Blöcken der Blockchain verglichen. Dabei kann es zu drei Fällen kommen (siehe auch Abbildung 5):

- Der passende Block ist das Ende (der neueste Block) der bestehenden Blockchain. Der Block wird an diese Blockchain, die auch *Main Chain* genannt wird, angehängt.
- Der passende Block ist nicht das Ende der Blockchain, sondern ein älterer Block, also ein Block, auf dessen Basis bereits neuere Blöcke erzeugt wurden. Auch dieser Fall ist für das Bitcoin-Protokoll normal, der Block wird an den alten Block angehängt und bildet eine sogenannte *Secondary Chain*.
- Der passende Block konnte nicht gefunden werden. Der Block wird im sogenannten Orphan Block Pool gespeichert bis der Elternblock empfangen wird und beide der Blockchain hinzugefügt werden können.

In allen der aufgeführten Fälle wird der Block an benachbarte Knoten weitergeleitet. Falls zwei gültige Blöcke etwa zum gleichen Zeitpunkt[4] ins Netzwerk eingespeist werden, kann es zu einem sogenannten Blockchain Fork kommen. Ein Teil des Netzwerks empfängt beispielsweise Block A zuerst und der andere Teil empfängt Block B zuerst. Letztendlich erhält jede Full Node beide Blöcke. Ein Teil des Netzwerks hat dann Block A als neuestes Main Chain Element und Block B als Secondary Chain Element eingetragen. Die anderen Knoten, die Block B zuerst erhalten haben, haben die Main und Secondary Chain exakt umgekehrt

[4] Tatsächlich ist dieser Zeitraum relativ groß. Solange ein Block noch nicht im gesamten Netzwerk verbreitet wurde, arbeitet ein Teil der Miner noch an der Suche nach dem alten Block (siehe Abschnitt 4). Bis zur Verbreitung des Blocks im gesamten Netzwerk kann es zu 'gleichzeitigen' Einspeisung von Blöcken kommen.

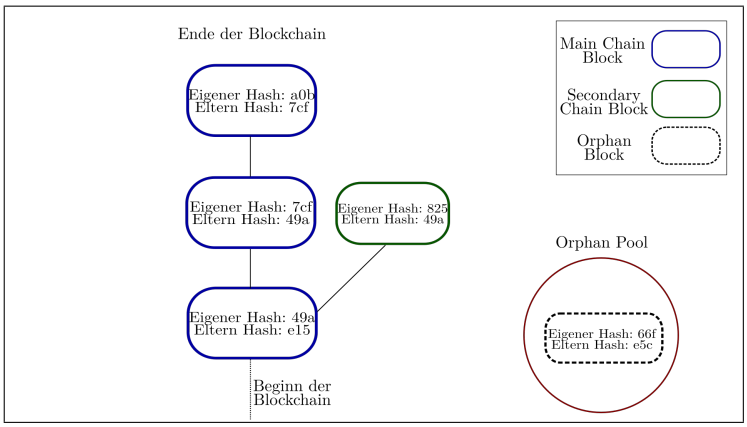

Abb. 5. Die verschiedenen Typen von Blockchains: Main Chain, Secondary Chain und Orphan Pool.

aufgebaut. Dieser Konflikt ist üblich und wird durch den nächsten Block, der ins Netzwerk eingespeist wird, gelöst. Basiert dieser neue Block auf Block A, dann werden alle Knoten die längere der beiden Ketten auswählen. Die längste Kette entspricht der Kette, in die die meiste Arbeit investiert wurde (*Proof of Work*), da jeder Block etwa der gleichen Menge Arbeit entspricht.

4 Mining

Der Begriff Mining lässt vermuten, dass das Mining primär für die Generierung neuer Bitcoins genutzt wird. Tatsächlich ist die Generierung neuer Bitcoins nur eine Belohnung neben den Gebühren für Miner. Dadurch werden Miner für ihre Suche nach einem neuen Block und die damit verbundene Arbeit belohnt. Für das Bitcoin-Protokoll dient das Mining hauptsächlich der Validierung und Aufzeichnung der Transaktionen auf der Blockchain. Durch die Aufzeichnung auf der Blockchain wird das sogenannte *Double-Spend-Problem* gelöst (siehe Abschnitt 2.5).

Beim Mining werden Transaktionen in Blöcke zusammengefasst. Dafür benötigt ein Miner Zugriff auf den Transaction Pool einer Full Node. Ein Teil der im Pool enthaltenen Transaktionen wird für den nächsten Block ausgewählt. Die Auswahl wird dabei über die Größe des Blocks und die enthaltenen Gebühren getroffen. Der Block darf die maximale Blockgröße nicht überschreiten und der Miner bevorzugt Transaktionen mit hohen Gebühren. Zusätzlich wird die sogenannte Coinbasetransaktion als erste Transaktion in den Block eingefügt. Die Coinbasetransaktion enthält die Gebühren und die generierten Bitcoins, welche auf eine private Adresse überschrieben werden. Anschließend müssen alle Header Felder entsprechend der Definition gesetzt werden und der *Proof-of-Work*-Algorithmus wird ausgeführt (siehe Abschnitt 2.4). Findet ein Miner einen Block

oder empfängt ein Miner den Block, den ein anderer Miner gefunden hat, beginnt der Miner die Arbeit am nächsten Block. Nur der Miner, der den Block gefunden hat, erhält die volle Belohnung. Es handelt sich beim Mining also um einen *Wettbewerb zwischen den Minern*[5].

Die Länge des Intervalls, in dem neue Blöcke der Blockchain hinzugefügt werden, soll konstant bei circa 10 Minuten liegen. Dabei handelt es sich um einen Kompromiss zwischen der schnellen Validierung von Transaktionen und der Verhinderung von Blockchain Forks. Um dieses Intervall sicherzustellen, wird die Zeit gesteuert, die ein Miner benötigt, um einen gültigen Block zu erzeugen. Durch den technologischen Fortschritt und die steigende oder sinkende Anzahl von Minern im Bitcoin-Netzwerk verändert sich die insgesamt zur Verfügung gestellte Rechenleistung (auch *Hashing Power* genannt). Entsprechend wird die benötigte Zeit zum Minen eines Blocks größer oder kleiner. Um den zuvor beschriebenen Kompromiss zu gewährleisten, muss der Schwierigkeitsgrad des Minings kontinuierlich an die Rechenleistung angepasst werden. Hierfür wird in jeder Full Node alle 2016 Blöcke der Schwierigkeitsgrad angepasst. Dafür wird die tatsächlich benötigte Zeit für das Mining der letzten 2016 Blöcke mit der gewünschten Zeit (20160 Minuten) verglichen. Ist die benötige Zeit kleiner/größer, wird der Schwierigkeitsgrad erhöht/gesenkt.

5 Konsens und Sicherheit

Um zu verstehen warum, die in den letzten Abschnitten erläuterten Verfahren notwendig sind, lohnt sich eine erneute Betrachtung der grundlegenden Ideen von Bitcoin. Bitcoin ist eine dezentralisierte digitale Währung, die ohne eine zentrale Autorität auskommen soll. Das Mining dient dabei der Validierung von Transaktionen und ist damit essentiell für die Sicherheit von Bitcoin. Diese Validierung beziehungsweise das Mining muss dabei ebenfalls dezentralisiert ausgeführt werden. Das Mining ist dabei ein möglicher Angriffspunkt, da jeder daran teilnehmen und Transaktionen rückgängig machen kann. Bitcoin nutzt das Proof-of-Work-Verfahren, um es Angreifern möglichst schwer zu machen, solche Änderung durchzuführen.

Durch den Schwierigkeitsgrad (die Erstellung eines Blocks benötigt ca. 10 Minuten) und die Verkettung von Blöcken (siehe Abschnitt 2.4) muss ein Angreifer viel Arbeit aufwenden, um die Blockchain zu verändern. Verändert ein Angreifer zum Beispiel den vorletzten Block in der Blockchain, so muss er auch den letzten Block neu berechnen, da das Netzwerk nur die längste Kette akzeptiert, also die Kette, in der die meiste Arbeit (Proof-of-Work) steckt. Die Wahrscheinlichkeit, dass der Angreifer aufholt, sinkt exponentiell unter der Voraussetzung, dass die ehrlichen Miner mehr Hashing Power besitzen [4]. Damit ist es dem Angreifer nicht mehr möglich, alte Blöcke der Blockchain zu verändern, da der Rechenaufwand zu groß wird.

[5] Oft arbeiten Miner in einem Mining Pool an der gemeinsamen Lösung des Problems und teilen die Gewinne entsprechend der zur Verfügung gestellten Rechenleistung auf.

Der Schlüssel zur Sicherheit von Bitcoin ist also die Hashing Power der ehrlichen Miner immer größer zu halten als die Hashing Power eines Angreifers. Im Idealfall würde jeder verfügbare Rechner nur mit dem Mining von Bitcoin beschäftigt sein. Ein Angreifer kann damit unmöglich mehr Rechenleistung erhalten. Um möglichst nahe an diesen Idealfall zu kommen, muss ein Anreiz geschaffen werden, ehrliches Mining zu betreiben. Dieser Anreiz ist die in Crefsec:mining erwähnte Belohnung. Sie sollte größer sein als die aufzuwendenden Energiekosten und den Gewinn, den ein gelungener Angriff mit sich bringt. Durch den Wettbewerb zwischen Minern (siehe Abschnitt 4) wird dieser Effekt noch verstärkt. Würde die Belohnung an alle beteiligten Miner gleichmäßig ausgezahlt, so würde es keinen Anreiz geben, die eigene Rechenleistung zu erhöhen. Durch den Wettkampf zwischen den ehrlichen Minern entsteht ein Wettlauf um mehr Rechenleistung, bei der ein Angreifer ebenfalls mithalten muss.

In Abschnitt 3.2 wird beschrieben, wie jeder Knoten im Netzwerk den längsten Block auswählt. Die längste Kette bedeutet durch das Proof-of-Work-Verfahren auch, dass die Kette durch die größte Menge Hashing Power erstellt wurde. Die Annahme dabei ist, dass die größte Menge Hashing Power durch das Belohnungsverfahren immer auf der Seite der ehrlichen Miner liegt. Auf diese Weise entsteht im Netzwerk Konsens über den aktuell gültigen Stand der Blockchain und somit darüber, welche Geldmittel bereits ausgegeben wurden und welche nicht. Es kann dabei, wie in Abschnitt 3.2 gezeigt, zu temporären Blockchain Forks kommen. Die Ursache hierfür könnte zum Beispiel ein Betrugsversuch sein oder auch gleichzeitig erstellte Blöcke. Im Falle eines Betrugsversuchs gelingt dieser nur, wenn wenige Blöcke (1-2) geändert werden müssen oder wenn die unehrliche Hashing Power über einen längeren Zeitraum größer ist als die Menge der ehrlichen Hashing Power.

6 Zusammenfassung

Die Miner in Bitcoin sind essenzieller Bestandteil, um die Blockchain aufzubauen und damit das *Double-Spend-Problem* zu lösen. Jeder ehrliche Miner stimmt mit seiner Rechenleistung für den korrekten Aufbau der Blockchain im Gegensatz zu anderen verteilten Systemen, bei denen pro IP-Adresse eine Stimme abgegeben wird. Der Mehrheitsentscheid ändert sich also von einem *Wer besitzt die Meisten IPs?*-System hin zu einem *Wer besitzt die größte Rechenleistung?*-System. Wer also die meiste Energie aufwenden kann, besitzt die Kontrolle über die Blockchain. Auf diese Weise wird das Bitcoin-Netzwerk weniger anfällig gegenüber Angriffen, da es für einen Angreifer schwieriger ist, eine große Menge Rechenleistung aufzubringen als eine große Menge IP-Adressen. Durch eine unabhängige Prüfung aller Blöcke und die Auswahl der längsten Kette wird Konsens im Netzwerk zwischen allen ehrlich agierenden Full Nodes hergestellt.

In Bitcoin wird also Sicherheit durch hohen Energieverbrauch (Mining) hergestellt. Dass der meiste Energieverbrauch auf der ehrlichen Seite liegt, wird durch die Belohnungen beim Mining sichergestellt. Allerdings ist die Rentabilität des Minings von äußeren Faktoren bestimmt. Dazu gehört der Wert (in konventio-

nellen Währungen) von Bitcoin, Elektrizitätskosten, Hardwarekosten und die Menge der Bitcoin Nutzer bzw. die Höhe der Gebühren, die bei einer Transaktion anfallen. Die Sicherheit von Bitcoin basiert also auf äußeren Faktoren, auf die das Bitcoin-Netzwerk selbst keinen Einfluss hat. Außerdem sind die Schäden für die Umwelt von Bitcoin nicht vernachlässigbar. Würde Bitcoin global und als einzige Währung eingesetzt werden, würden die globalen CO_2-Emissionen um 2,1 % ansteigen. Dabei handelt es sich nur um Emissionen durch die Nutzung von Elektrizität. Der Wert läge noch höher, wenn Emissionen durch Kommunikationsinfrastruktur und Emissionen bei der Herstellung benötigter Hardware ebenfalls betrachtet werden. Demgegenüber sind die Emissionen eines vergleichbaren, global agierenden konventionellen Währungssystems vernachlässigbar [3].

Das *Proof-of-Work*-System eignet sich also, um das *Double-Spend-Problem* zu lösen unter der Bedingung, dass die äußeren Faktoren stabil bleiben. Werden mit Bitcoin große Beträge überwiesen, sollte zudem gewartet werden, bis die Transaktion durch einige zusätzliche Blöcke (circa 6 Blöcke) bestätigt wurde, da dies die Wahrscheinlichkeit einer Änderung gegen Null gehen lässt (siehe Abschnitt 5). Die Frage der Umweltverträglichkeit eines solchen Systems bleibt jedoch offen. Hierfür müssen im Falle einer flächendeckenden Übernahme eines Systems wie Bitcoin Lösungen für eine Reduktion des Energieverbrauchs gefunden werden.

Referenzen

1. Bitcoin Core (2021), `https://github.com/bitcoin/bitcoin`
2. Antonopoulos, A.M.: Mastering Bitcoin: Programming the Open Blockchain, 2nd edn. (2021), `https://github.com/bitcoinbook/bitcoinbook`
3. Becker, J., Breuker, D., Heide, T., Holler, J., Rauer, H.P., Böhme, R.: Can We Afford Integrity by Proof-of-Work? Scenarios Inspired by the Bitcoin Currency, pp. 135–156. Springer Berlin Heidelberg, Berlin, Heidelberg (2013), `https://doi.org/10.1007/978-3-642-39498-0_7`
4. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system (2009), `http://www.bitcoin.org/bitcoin.pdf`

Alle Links wurden am 17. Februar 2021 das letzte Mal aufgerufen.