Um estudo do Blockchain e sua aplicação em criptomoedas

C. V. Rizzo, M. F. Rocha, N. A. Calmon

Resumo — O uso e estudo de criptomoedas segue um crescente desde a criação da mais conhecida no ramo, o Bitcoin. A partir dele, e do desenvolvimento do seus sistema baseado em blockchain, que diferentes problemas computacionais, incluindo de aplicações em sistemas distribuídos, começaram e ser resolvidos. Neste artigo é mostrado amplamente o funcionamento do sistema blockchain, com um foco maior em sua aplicação em criptomoedas. Existe um tópico especial para a primeira a propor seu uso, seguindo de uma comparação, do mesmo, com características de sistemas distribuídos e, por último, alguns mecanismos, que surgiram após sua criação, para aperfeiçoar aspectos em sua aplicação.

***Palavras Chave* — Criptomoeda, Cryptocurrencies, Blockchain, Sistemas Distribuídos, Peer-to-peer Network, Bitcoin, altcoin.**

I. INTRODUÇÃO

Para que houvesse troca de bens juntamente com um mecanismo de reserva de valor criou-se a moeda, que pode ser caracterizada por ser uma ferramenta financeira[1, pp. 432]. Com ela é possível a transação monetária que se configura pela compra e venda de qualquer bem que tenha um valor agregado.

De modo tradicional, hoje, uma transação é constituída de duas partes sem comunicação direta, sendo necessário a existência de pelo menos um intermediário confiável como o banco central, além de bancos comerciais e de investimentos, corretoras de valores, fundos de investimentos, fundos de pensão, bolsas de valores e companhias de seguro, que farão a gestão dessa operação[2][3, pp. 426]. Apesar desse sistema ser eficiente para a maioria das transferências, ele ainda é dependente de confiança, onde cada lado deve acreditar que os acordos pré estabelecidos serão cumpridos, mesmo circulando pelos mediadores, além de essas mediações estabelecerem um valor mínimo para executar as operações, encarecendo as mesmas[4, pp. 1].

Uma alternativa prática começou a ser explorada em 2008 a partir da crise financeira, que emergiu discussões sobre novas formas de emissão, distribuição e uso do dinheiro[1, pp. 431], para que em vez de confiança, as transações fossem baseadas em criptografia, com a criação das criptomoedas descentralizadas utilizando o conceito de blockchain, que segundo Daniel Kraft [5, pp 397] "*... all transactions are*

Tabela 1 [7, pp. 2092]

	Banking Model	Bitcoin
regulation/oversight	central bank	consensus
transaction verification	centrally	consensus
money creation	loans	mining
money supply	virtually unlimited	finite supply
value of money	exchange rate	proof of work, supply and demand, trust
money transfer	mediated, reversible	direct, non-reversible
privacy	implementation-dependent	somewhat anonymous
fees	account charge, transaction charge	virtually constant transaction charge
transaction delay	theoretically instantaneous, practically in the order of days	in the order of tens of minutes

performed and validated by a peer-to-peer system, where each node is "equal" and none has any special authority", que diz que as transações passam a ser diretas sem a necessidade de nenhum órgão centralizador.

As diferenças entre o modo tradicional - modelo bancário - e o baseado em blockchain - modelo usado pelo Bitcoin -, que foram citados acima, podem ser vistas de maneira mais clara e completa na tabela 1.

O restante deste artigo será estruturado da seguinte forma. A seção II mostra um histórico das criptomoedas desde sua criação até o desenvolvimento do Bitcoin. A seção III se trata de um estudo amplo sobre o Bitcoin. Na seção IV é abordado um comparativo do blockchain com as características de sistemas distribuídos. Finalmente, a seção V mostra mecanismos do blockchain que surgiram para seu aperfeiçoamento, aplicados em outras criptomoedas.

II. HISTÓRICO

A demanda por uma forma de *e-commerce* mais ágil tem impulsionado o surgimento de diversos sistemas de pagamento online ao longo da última década[6, pp. 86]. Além das formas já tradicionais de cartão de crédito surgiram os chamados "métodos alternativos de pagamento", que incluem carteiras eletrônicas (eWallets), como *Paypal* e *Google Checkout,* e sistemas de transferência de dinheiro, os quais, se baseiam em moedas fiduciárias existentes, o que não ocorre com as moedas digitais[6, pp. 86]. Essas tiveram seu primeiro vislumbre no início dos anos 80, mas levaram mais de um quarto de século para que uma solução totalmente distribuída se tornasse realidade, ou seja, um sistema livre de uma entidade central de autoridade como um banco fosse implementado[7, pp. 2084].

O Bitcoin e as moedas digitais similares são chamadas de criptomoedas devido aos algoritmos de segurança que elas utilizam, os quais estão intimamente relacionados com algoritmos criptográficos [8, pp. 83]. O gênesis das criptomoedas foram os cypherpunks vindo de um tipo de amor à internet e suas possibilidades. Eles discutiam sobre a possibilidade de criação de uma moeda que fosse anônima ou que pudesse ficar anônima usando a criptografia[9].

Apesar de David Chaum não ser um cypherpunk, há registros de que uma das primeiras moedas virtuais é o DigiCash criada pelo mesmo e seus colegas do centro de pesquisas nacional em matemática e ciência da computação CWI (Centrum Wiskunde & Informatica), na Holanda[10, pp. 97]. Esta nova moeda tinha um conceito de assinaturas anônimas, que é um modelo de sistema de pagamento em dinheiro, o qual provê um alto nível de anonimato, além de não ser facilmente rastreável[11, pp. 33]. Não apenas possui as características acima como ela também utiliza de um complexo processo de encriptação para garantir a segurança do seu sistema de pagamentos[12].

Chaum, sentiu a necessidade de procurar alguma instituição já existente, como bancos ou governo, para ajudá-lo em seu projeto, e com isso, sua relação com o movimento cypherpunk foi quebrada, uma vez que ele servia de inspiração para o mesmo, desencadeando o desânimo de seus seguidores e fazendo com que os projetos relacionados a eles fossem engavetados [9].

As primeiras tentativas de criação de um sistema distribuído ainda possuíam a figura central de um banco, como o Digicash de David Chaum, apesar de, implementações posteriores como B-money de Wei Dei, Karma de Vivek Vishnumurthy, Sangeeth Chandrakumar e Emin Gun Sirer, RPOW de Hal Finney e Bit Gold de Nick

Szabo (que possuem suas principais diferenças vistas na tabela 2) introduzirem a ideia de *proof of work*, que consiste, em um quebra-cabeça que deve ser resolvido e que demanda um certo esforço computacional, evitando assim ataques do tipo Sybil - que é o ato de criar instâncias de confirmação falsas por um usuário mal intencionado, a fim de possuir a maioria das confirmações da rede -. Com isso, esse sistema torna confirmações falsas algo difícil de se conseguir, e assim, faz com que o poder computacional seja mais relevante do que a quantidade de usuários mal intencionados, adicionando, dessa forma, mais segurança e removendo a necessidade de uma instância central como autoridade máxima, mesmo ainda precisando da mesma uma para manter registros de propriedade[7, pp. 2084]. Esse conceito será mais explorado na seção 3.

A partir da crise financeira de 2008, que ocorreu no centro do capitalismo e que gerou uma falta de confiança não apenas nos bancos e no mercado, mas também na economia norte-americana como um todo[13], desencadeou-se discussões em torno das deficiências dos instrumentos e mecanismos do sistema financeiro trazendo à tona várias práticas alternativas e complementares, tanto em escala local quanto global[1, pp. 431], e assim, os experimentos da década de 90 voltaram a ser relevantes.

Tabela 2 - Comparação entre moedas [7, pp. 2118]

Moeda Virtual	Diferenças
B-Money	Mineração por recompensa proporcional a dificuldade do *proof of work*; Requer um canal broadcast [14].
Bit Gold	*Proof of work* baseada em uma corrente de assinatura digital [15].
Karma	Sistema distribuído de moedas ainda mantido por um banco [16, pp. 1].
RPOW	Esse sistema utiliza *hashcash* como token para a proof of work [17].

A partir disso, surgiu o Bitcoin, que será visto mais detalhadamente na seção 3, sendo considerado uma resposta direta aos problemas profundamente enraizados no sistema monetário que foram revelados pela crise, por ser uma contra proposta inovadora e por apresentar formas de se construir um novo sistema de pagamentos baseados em moedas digitais[18, pp. 18]. Com seu enorme sucesso, o Bitcoin atraiu os mais diversos grupos de usuários, os quais podem ser classificados como geeks, investidores, idealistas e criminosos[19, pp. 1030].

III. BITCOIN

Eis que em 2008 Satoshi Nakamoto, um pseudônimo, pública o artigo Bitcoin: A Peer-to-Peer Electronic Cash System, descrevendo sua criação como: "*A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution*"[4, pg 1], ou seja, uma versão descentralizada de criptomoeda - sem a necessidade de um órgão fiscalizador do dinheiro eletrônico.

Como foi visto anteriormente, as moedas virtuais ainda utilizavam alguma figura central, ou seja, não eram completamente distribuídas, diferente do Bitcoin. Diante disto pode surgir um questionamento como abordado por Florian Tschorsch e Björn Scheuermann[7, pp. 2086], "*So, how can we eliminate the central bank? Bitcoin solves this in a very pragmatic way: in a sense, everyone is the bank*". Ou

seja, cada participante mantém uma cópia dos dados que ficariam armazenados no banco central, e dessa forma, se torna desnecessária a existência desse tipo de entidade. Logo, esse sistema pode ser considerado um *ledger* (livro-razão) distribuído, que reflete todas as transações e seus envolvidos. No Bitcoin é o *blockchain* que assume essa função, permitindo a ocorrência de diversas transações[7, pp. 2086][9].

Tradicionalmente era o banco quem assumia o papel de intermediário confiável nas transações eletrônicas financeiras, já o Bitcoin traz um sistema de pagamento eletrônico baseado em criptografia no lugar da confiança, permitindo que duas partes quaisquer fizessem transações diretamente sem a necessidade desse mediador de segurança [4, pp.1].

Uma das principais funções das instituições financeiras era assegurar que não ocorreria o problema de *double-spending* (gasto duplo) definido por Gerald Dwyer em The economics of Bitcoin and similar private digital currencies[8, pg. 82] como "*a digital representation of currency requires that it not be possible to create multiple copies and spend the same digital currency two or more times*", ou seja, não fazer mais de uma transação com uma mesma moeda pois se não ela perde seu valor [9].

No modelo bancário esse problema é resolvido com o uso de números de série controlados pelo banco e pela proibição de processamentos concorrentes de transações, forçando uma ordenação total destas [7, pg. 2086]. Um problema com essa proposta é ressaltada por Nakamoto [4, pg. 2] "The problem with this solution is that the fate of the entire money system depends on the company running the mint, with every transaction having to go through them, just like a bank".

Com o uso do Bitcoin o problema descrito acima não é solucionado por apenas uma pessoa ou uma instituição, mas sim por toda a rede distribuída, pois cada usuário está ciente de todas as transações, permitindo assim, que toda a rede verifique a legitimidade das mesmas, de modo que, o gasto duplo seja percebido por outros participantes. A transação somente será aceita se a maioria dos participantes concordar com a sua existência [6, pp. 86][7, pp. 2086].

Os sistemas de quórum, no qual, aceitam a possibilidade de informações incorretas e a existência de entidades maliciosas em um ambiente distribuído, introduzem o conceito de voto. Enquanto a maioria de qualquer subconjunto de participantes (quorum) escolhido for honesta, é garantindo a permanência do estado correto através de eleição [7, pp. 2084].

Para que haja a comunicação entre todos os nós, o Bitcoin usa uma rede peer-to-peer que utiliza timestamps nas transações por meio de hashes, dentro de uma cadeia contínua de *proof-of-work* [4, pp. 1]. Dado isso, para melhor compreensão, segue abaixo uma explicação de cada um desses conceitos:

A. Proof of Work (PoW)

Antes de explicar o que é a *proof-of-work* e como esta irá garantir a consistência das informações salvas na Blockchain do Bitcoin, é necessário entender o que é um *hash*.

Um *hash* é uma transformação da informação original e sua função(H), portanto, irá pegar uma mensagem M com um tamanho arbitrário e produzir um valor *hash* h, o qual h = H(M).

O Bitcoin usa um subconjunto das funções *hash*, chamadas de *one-way hash*, as quais são consideradas unidirecionais, apenas podendo ser revertidas a troco de um

custo verdadeiramente alto. As principais características do *one-way hash* são [8, pg. 83]:

1) Dado um M, é fácil computar h;
2) Dado um h, é difícil computar M, onde H(M) = h;
3) Dado um M1, é difícil achar outra mensagem M2, a qual H(M1) = H(M2).

O processo de trabalhar em novos *hashes* no Bitcoin é realizado por usuários especializados, os mineradores [20, pp. 50], os quais recebem uma recompensa, em valor de Bitcoin, que é gerada pelo sistema e transferida em um tipo especial de transação, a chamada *coinbase transaction*, para o primeiro que encontrar o valor do *hash* de um bloco [7, pp. 2089] [4, pp. 4]. Esse incentivo, característico do Bitcoin, é descrito por Nakamoto como "*The steady addition of a constant of amount of new coins is analogous to gold miners expending resources to add gold to circulation*" [4, pp 4], sendo os recursos gastos, neste caso, tempo de CPU e eletricidade [4, pp.4]. É desta analogia que deriva a terminologia de minerar Bitcoins e aqueles que realizam esta tarefa, a troco de moedas, os mineradores.

Novas moedas são criadas ao terem um problema computacional resolvido, após encontrarem a resposta do problema proposto pelo *proof of work* [8, pp. 83], o qual será utilizado para encadear mais um bloco na cadeia que forma o *blockchain*.

Como Nakamoto explica no *whitepaper* do Bitcoin [4, pp. 3], a *proof of work*, descrita por ele é similar ao Hashcash do Adam Back. Esta envolve a procura por um valor de *hash*, utilizando de uma função criptográfica como SHA-256, e incrementando um *nonce* até que um *hash* com uma certa quantidade de bits 0 seja encontrada.

O trabalho computacional demandado para encontrar o valor satisfatório é exponencial no número de bits 0 demandado, porém para sua verificação basta o cálculo de um único *hash* [4, pp. 3]. Com isso a dificuldade dos mineradores não é computar o *hash*, que é fácil, mas encontrar o *hash* adequado através de uma busca *brute-force* [8, pp. 83] [5, pp. 398].

Para proteger a integridade do *ledger* cada bloco de transações possuirá um único *hash*, que será derivado da informação contida no mesmo e o valor do *hash* do bloco anterior, criando assim uma corrente de blocos que não pode ser alterada sem refazer o *proof-of-work*.

Portanto a função de *hash* utilizada pelo bitcoin é a seguinte [21, pp. 133]:

H (N || P_hash || Tx || Tx || ... Tx) < Target

Onde *N* é o *nonce* que varia para alterar os valores do *hash*, *P_hash* é o *hash* do bloco anterior, *Tx* representa as transações no bloco e *Target* é a dificuldade do *hash* (chamada de dificuldade da rede) que é imposta pela rede a ser satisfeita. Isto significa que o *hash* que deve ser encontrado pelos mineradores deve ser menor que o imposto pelo *Target* para poder ser considerado válido [21, pg. 133].

Novos blocos são adicionados a *blockchain* aproximadamente a cada 10 minutos, com a dificuldade da rede sendo ajustada dinamicamente a cada 2016 blocos de forma que mantenha esta taxa de adição constante[21, pg. 130].

A dificuldade da rede é calculada usando a seguinte equação:

Target=Previous target * Time/2016 * 10min

Target representa a dificuldade, *Previous target* é o antigo valor do *target* e *time* é o tempo que foi gasto até gerar os 2016 blocos anteriores. Portanto a dificuldade da rede basicamente significa o quão difícil vai ser para os mineradores

encontrarem o *hash* do próximo, ou seja, o quão difícil o quebra cabeças criptográfico está no momento [21, pp. 131].

A utilização do valor *hash* do bloco anterior força os possíveis invasores a ter de refazer todo o trabalho computacional, já realizado pelos mineradores, a partir do bloco que estes tiverem a intenção de fraudar (removendo alguma transação ou alterando o destino de alguma moeda por exemplo) [20, pp. 50][4, pp. 4]. Este tipo de fraude se torna ainda mais complicada de ser realizada, uma vez que o *proof of work* do Bitcoin determina que os nós que compõem sua rede apenas irão aceitar como válida a cadeia de blocos mais longa, ou seja, caso exista conflito, o *branch* (ramo) que contém mais "provas de trabalho" será considerado aquele que com o estado verdadeiro do *ledger*. Logo mineiros sempre irão adicionar seus novos blocos na cadeia mais longa conhecida [5, pp. 398].

A grande vantagem deste sistema está no fato da verificação desta resposta poder ser realizada pelos demais com baixo custo computacional, enquanto para encontrá-la, um grande esforço é demandado. Sua dificuldade está sujeita a ser incrementada ao longo do tempo, com um eventual limite no número de moedas criadas (21 milhões de bitcoins) [8, pp. 83].

Outro problema solucionado pelo *proof-of-work* é o da decisão da maioria, o consenso em redes distribuídas. Como dito anteriormente, os nós da rede irão aceitar a cadeia mais longa de blocos como verdadeira, portanto, diferente de um modelo baseado em um-IP-um-voto, o qual poderia ser subvertido por qualquer usuário que pudesse alocar IPs suficientes para ter a maioria da rede, o sistema do *proof-of-work* é essencialmente um-CPU-um-voto. Logo a maioria é determinada pela maior "prova de trabalho" e portanto aquela cadeia de blocos que teve mais esforço computacional investido. Como ressalta Nakamoto [4, pp. 3] "*If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest and outpace any competing chains.*". Desta forma, a rede continuará honesta, uma vez que a maior parte do poder computacional for composta de usuários honestos [4, pp. 3].

B. Timestamp

O bloco da Blockchain do Bitcoin possui um campo *timestamp*, o qual segundo o livro Mastering Blockchain diz que "*This field contains the approximate creation time of the block in the Unix epoch time format. More precisely, this is the time when the miner has started hashing the header*" [21, pp. 128], guardando, portanto, o tempo pelo ponto de vista do minerador que encontrou o *hash* daquele bloco.

Segundo Nakamoto [4, pg. 2] "*The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash.*". Desta forma, a Blockchain funciona como um grande *timestamp*, o qual cada *timestamp* está vinculado ao anterior pela ligação entre os hashes (Figura 1), servindo portanto como uma prova da sequência dos eventos que ocorreram [4, pg. 1 e 2].

Ter conhecimento da ordem que cada ação ocorreu é essencial em um sistema de transações e como o mecanismo de *proof-of-work* do Bitcoin reajusta sua dificuldade para que a geração dos blocos sempre leve cerca de 10 minutos, este pode ser considerado um serviço distribuído seguro de *timestamp* [7, pp. 2116].

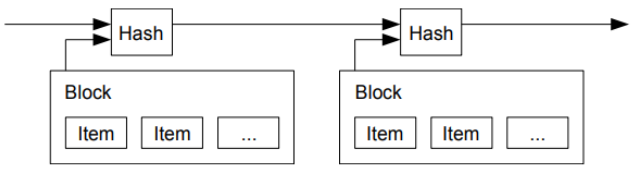

Figura 1 [4, pp. 2]

C. Transações

As transações são o coração do sistema do Bitcoins. Elas representam a transferência de um valor de um endereço para outro, podendo ser simples, como a transferência de alguns bitcoins ou complexas, com vários endereços de entrada e saída [21, pp. 19, 118].

Usuários de Bitcoin utilizam carteiras (*wallets* em inglês) para poderem acompanhar os seus saldos, enviar e receber quantias de Bitcoins. Essas, apesar de terem este nome, não contém de fato Bitcoins, podendo ser comparado a um programa de planilha eletrônica que mantém o controle do saldo do usuário.

Todos os Bitcoins estão associados a um endereço, que se relaciona a uma carteira, o qual é o nome de uma chave pública (*public key*). A criptografia utilizada na chave pública é essencial para que se possa conseguir guardar e rastrear o saldo dos indivíduos na Blockchain. Esta, juntamente com uma chave privada (private key) são utilizadas para criptografar e descriptografar mensagens, o que é crucial para conseguir definir quando uma transação é de fato válida. O endereço para o qual um Bitcoin está sendo enviado é uma encriptação da chave pública do receptor, ou a própria chave [8, pp. 84].

A assinatura digital, como pode ser visto na Figura 2, é uma encriptação feita utilizando a chave privada do usuário que está enviando, podendo ser descriptografada utilizando a sua chave pública (que é conhecida), fazendo com que o destinatário da transação possa ser conhecido pela rede e o remetente possa ser verificado sem revelar sua chave privada[8, pp. 84].

Dessa forma o que a carteira digital mantém é o controle da chave pública e da chave privada, a qual não deve ser perdida, pois sem ela não é possível produzir a assinatura digital para que alguma transferência seja feita, e assim, os Bitcoins que estão vinculados àquele endereço serão perdidos [8, pp. 84].

As chaves privadas são basicamente números de 256 bits gerados aleatoriamente [21, pg. 118]. O Bitcoin utiliza ECC (Elliptic Curve Cryptography) baseado na SECP256K1 padrão como algoritmo de criptografia, para gerar as chaves públicas através de uma *private key*. Esta é basicamente uma coordenada x e y (ambas de tamanho 32 bits) em uma curva elíptica, sendo representadas com o prefixo 04 em hexadecimal.

Uma versão compactada dessa chave começou a ser utilizada como padrão, reduzindo-a de 65 para 33 bytes, e podendo ser identificadas por vários prefixos, como 0x02, 0x03 e 0x04, de acordo com seu tipo de compactação[21, pp. 115].

As moedas, no sistema do Bitcoin, são, na verdade, abstrações das transações gravadas no Blockchain, assim como Satoshi Nakamoto define [4, pp. 2]: “*We define an electronic coin as a chain of digital signatures. Each owner transfers the coin to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin*”. Logo, as transações se ligam umas com as outras formando uma corrente de hashes (Figura 2), com a qual é possível verificar a legitimidade de uma transação percorrendo-as na Blockchain para determinar o verdadeiro dono de alguma moeda, bastando para isso, encontrar a transação mais recente da mesma[7, pp. 2084][20, pp. 50].

Para verificar se algum destes não cometeu o gasto duplo, sem utilizar alguma entidade central para isto, é necessário que toda a rede tenha conhecimento de todas as transações que ocorreram e portanto estas devem ser anunciadas publicamente [4, pg. 2].

Cada registro de transação (Figura 2), portanto, conterá uma chave pública (public

key) e para cada transação de Bitcoin, o dono atual, utiliza sua chave privada (private key) para se legitimar como verdadeiro dono da quantia em moedas, e então, envia uma instrução de transação encriptada com esta chave. O sistema irá guardar a transação, a qual terá a chave pública do receptor, que será o novo dono da quantia, em um novo bloco[8, pp. 84].

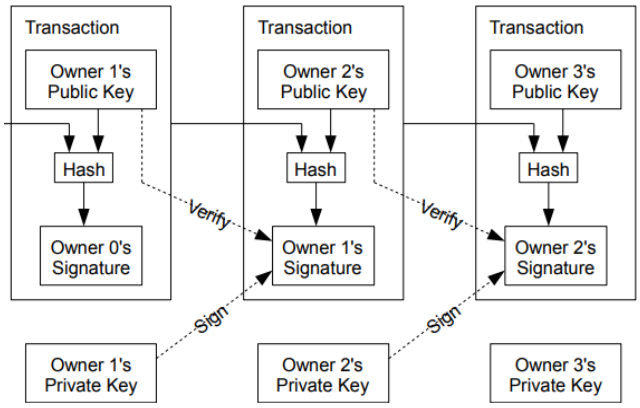

Figura 2 [4, pp. 2]

O Bitcoin utiliza uma linguagem baseada em pilha chamada Script para descrever como eles podem ser gastos e transferidos. Ela foi intencionalmente projetada para não ser Turing completa, tornando-a mais fácil de lidar e evitando alguns efeitos não desejados. O uso desta linguagem dá às transações muito mais flexibilidade e um certo grau de programabilidade em exatamente o que elas irão fazer.[7, pp. 2089].

A quantia de Bitcoins que será transferida para um dado endereço é sempre descrita no *output* da transação, como mostra a Figura 3, e esse é especificado por um *script*. Operações que serão realizadas constituem o chamado *scriptPubKey*. Um *script* espera uma certa quantidade de argumentos, o chamado *scriptSig* [7, pp. 2089].

Cada *input* em uma transação de bitcoins está ligada á um *output* anterior, sempre utilizando todas as moedas do *output* referenciado. Com isso, cada *output* só pode ser referenciado uma única vez, pois caso haja uma outra tentativa, isto caracterizaria um *double spending* e portanto é totalmente proibido[22, pp. 3].

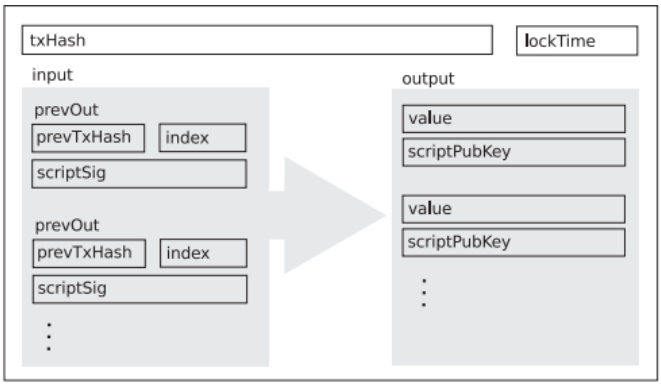

Figura 3 [7, pp. 2089]

Existem diversos *scripts* disponíveis no Bitcoin para lidarem com os mais diversos tipos de transferência de valor, de uma fonte para um destino, variando desde os mais simples até os mais complexos, dependendo da necessidade que a transação apresente. O *script* que é, provavelmente, o mais comum e essencial entre todos é o “Pay-to-PubKeyHash” (P2PKH), servindo para, basicamente, transferir moedas de um ou mais endereços de origem para um único destino. Seu formato pode ser visto abaixo [21, pp. 121, 122] [7, pp. 2089]:

ScriptPubKey: OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG

ScriptSig: <sig> <pubKey>

Neste script de bloqueio, ou scriptSig, P2PKH é necesário uma chave pública (pubKey), e uma assinatura criptografada (sig) gerada a partir da chave privada da transação, comprovando assim, sua posse sobre a mesma[7, pp. 2089, 2090][22, pp. 3].

Recompensas são cobradas pelos mineiros para que ocorra realização do

trabalho computacional, como em validar a transação e incluí-la em um bloco. Essas taxas são calculadas pela subtração da soma total dos valores no *input* da transação, com o total de *outputs*. Este incentivo funciona como uma gorjeta paga pelos usuários do Bitcoin para incentivar e encorajar os mineradores a continuarem trabalhando na rede.

Como Nakamoto determina em [4, pg. 4] “If the output value of a transaction is less than its input value, the difference is a transaction fee that is added to the incentive value of the block containing the transaction”, onde a inclusão destas recompensas não são nem fixas e nem obrigatórias no protocolo do Bitcoin. Mineradores irão optar por coletar primeiramente aquelas que possuem as maiores diferenças entre *input* e *output*, porém isto não significa que uma transação sem gorjeta nunca irá ser processada, ela irá, porém pode levar um tempo muito grande até que isto ocorra[21, pp. 125].

O ciclo de vida de uma transação é descrita por [21, pp. 118-119] como os seguintes passos abaixo:

“1) A user/sender sends a transaction using wallet software or some other interface.

2) The wallet software signs the transaction using the sender's private key.

3) The transaction is broadcasted to the Bitcoin network using a flooding algorithm.

4) Mining nodes include this transaction in the next block to be mined.

5) Mining starts once a miner who solves the Proof of Work problem broadcasts the newly mined block to the network.

6) The nodes verify the block and propagate the block further, and confirmation starts to generate.

7) Finally, the confirmations start to appear in the receiver's wallet and after approximately six confirmations, the transaction is considered finalized and confirmed. However, six is just a recommended number; the transaction can be considered final even after the first confirmation. The key idea behind waiting for six confirmations is that the probability of double spending is virtually eliminated after six confirmations.”

D. Rede

Como já dito acima o Bitcoin possui um sistema distribuído, ou seja, sem a necessidade de um intermediário, mas precisa da mediação de terceiros para validar e certificar todas as transações, sendo para isso, necessário o uso de uma rede *peer-to-peer* para fazer essa comunicação entre os usuários [6, pp. 86]. O próprio criador do Bitcoin define a moeda como “*a purely peer-to-peer version of electronic cash*”[4, pp.1].

Uma rede *peer-to-peer* é organizada como um conjunto de nós em uma rede conectada auto-organizada, onde alguns ou todos os nós podem atuar como clientes e servidores. Os nós são conectados uns com os outros, mas não necessariamente a todos outros nós da rede, sendo possível formar pares dado um nó e seu adjacente. Seu uso é o mais pertinente por ser mais resiliente a ataques ou em casos de problemas em algum nó específico [8, pp. 82]. Sendo assim, a estrutura de comunicação do Bitcoin vai utilizar de uma rede peer-to-peer baseada em TCP (Transmission Control Protocol) [7, pp. 2098], onde várias mensagens podem ser enviadas em uma mesma conexão [23, pp. 48].

Nakamoto estabeleceu alguns passos que devem ser seguidos para que a rede do Bitcoin funcione, sendo eles:

“1) New transactions are broadcast to all nodes.

2) Each node collects new transactions into a block.

3) Each node works on finding a difficult proof-of-work for its block.

4) When a node finds a proof-of-work, it broadcasts the block to all nodes.

5) Nodes accept the block only if all transactions in it are valid and not already spent.

6) Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash."

Na rede do Bitcoin ao se fazer um broadcast, utilizando alguma abordagem de inundação para que a informação seja rapidamente disseminada aos nós, as novas transações não precisam chegar necessariamente em todos os participantes[7, pp. 2098]. Desde que muitos sejam atingidos e entrem em um consenso, as transações acabam sendo incluídas em um novo bloco na cadeia. Essa rede também é tolerante a falhas e caso um participante não receba um bloco ele pode requisitá-lo depois quando receber um bloco subseqüente e descobrir que está faltando uma parte da cadeia[4, pp. 3-4].

Tabela 3 - Estrutura de um bloco [21, pp. 127-128]

Nome	Descrição
Block header	Este inclui campos de cabeçalho de bloco descrito na próxima seção
Contador de transações	O campo contém o número total de transações no bloco, incluindo a transação coinbase
Transações	Todas as transações em um bloco

Tabela 4 - Estrutura de um block header [21, pp. 127-128]

Nome	Descrição
Versão	O número da versão do bloco que determina as regras de validação de bloco a seguir
Hash do cabeçalho do bloco anterior	Este é um hash duplo SHA256 do cabeçalho do bloco anterior
Hash da raiz da árvore de Merkle	Este é um hash duplo SHA256 da árvore merkle de todas as transações incluídas no bloco
Timestamp	Este campo contém o tempo de criação aproximado do bloco no formato de hora da época do Unix. Mais precisamente, este é o momento em que o minerador começou a fazer o cabeçalho (o tempo do ponto de vista do mineiro)
Target de dificuldade	Esse é o alvo da dificuldade do bloco
Nonce	Este é um número arbitrário que os mineradores mudam repetidamente para produzir hashes diferentes e que atenda ao limite de dificuldade

Muitas dessas características só puderam ser implementadas por causa da existência do blockchain, que frequentemente é usados de forma intercambiável com o Bitcoin, mas não sendo os mesmos. O Bitcoin tem um blockchain, e este será abordado mais abordado a seguir.

IV. BLOCKCHAIN

A tecnologia do blockchain tem atraído a atenção como sendo a base do Bitcoin, porém é capaz de se estender para bem além dessa aplicação, permitindo que tecnologias antigas sejam melhoradas e que haja existência de novas aplicações, que antes desta não eram possíveis [25, pp. 15].

Segundo o artigo Blockchain Beyond Bitcoin escrito por Sarah Underwood [25, pp. 1], "*Blockchain is expected to revolutionize industry and commerce and drive economic change on a global scale because it is immutable, transparent, and redefines trust, enabling secure, fast, trustworthy, and transparent solutions that can be public or private.*", sendo assim, uma inovação transformadora com diversos benefícios que pode revolucionar em grande escala, os mais variados setores.

Por ser um sistema descentralizado, esses benefícios podem ser relacionados aos de um sistema distribuído e vistos a seguir.

A. Transparência

O armazenamento de transações se dá por uma árvore de Merkle, que será explicada no item C deste mesmo tópico, que relaciona as mesmas aos seus hashes e consequentemente ao hash do cabeçalho do bloco onde elas estão armazenadas.

Um utilizador comum, caso deseja saber se sua transação foi aprovada, deve solicitar à rede o último bloco processado. Sabendo desse bloco, o usuário pode comparar o hash da sua transação com os hashes daquele bloco. Caso sua transação não esteja neste, ele pode verificar os bloco anteriores, um a um[4, pp. 5].

Quando o usuário encontra o bloco, que referencia ao hash da transação que está sendo procurada para conferência, é possível determinar que essa foi confirmada. Assim, não é necessário acessar a transação em si, e nem as informações contidas nela para determinar seu status, fazendo com que haja transparência de acesso[26, pp. 08].

Outro tipo de transparência existente, é o de localização, que esconde de onde veio o bloco de transações, bem como quantos são e quem são os mineradores que realizaram esse processo. Como uma transação é enviado por *broadcast* na rede para que um mineiro possa processar suas informações, não há como saber, e nem é necessário, qual o mineiro que realizou esse procedimento[4, pp. 1].

B. Privacidade

Existe um consenso de que é possível linkar as informações contidas na *blockchain*, fazendo com que este não seja tão privado quanto assume ser e revelando que esse é provavelmente o mais transparente sistema de trocas já criado. Porém esta transparência do sistema é um aspecto fundamental para que este possa ter suas transações verificáveis [7, pp. 2107]. Como o artigo [7, pp. 2107] define, "*The blockchain is a huge record, which enables everybody to evaluate all transactions*", e é exatamente essa transparência que o *blockchain* possui como característica, que o torna mais atraente no setor financeiro, pois esta permite que as transações que estão ocorrendo possam ser monitoradas em tempo real[25, pp. 16].

Apesar desse aspecto de mostrar publicamente todas as transações do *ledger* parecer ser um aspecto de transparência que falta nos intermediários financeiros tradicionais, o único propósito desta é o de evitar falsificações, não tendo nenhum efeito de empoderamento dos usuários[18, pp. 30].

O artigo [6, pp. 86] ressalta: "*Bitcoin has the unintuitive property that while the ownership of money is implicitly anonymous, its flow is globally visible*", reforçando a ideia inicial de que esse *flow* de transações pode linkar com usuários, caso alguma identidade seja revelada.

Identidades na rede geralmente são pseudônimos, os quais, apesar de, não se ligarem explicitamente a pessoas ou organizações do mundo real, todas as transações realizadas por estes são completamente transparentes. Essa combinação incomum que causa uma

confusão considerável sobre a natureza e consequências da anonimidade que o Bitcoin provê[6, pp. 86], existindo, como Meiklejohn et al explicita, um consenso de que a combinação de pagamentos anônimos, escalonáveis e irrevogáveis seja extremamente atrativo para criminosos envolvidos em lavagem de dinheiro ou fraudes[6, pp. 86].

Essa anonimidade do Bitcoin se dá pelo uso das chaves públicas às quais servem de pseudônimo e não possuem limite por usuário, como [6, pp. 87] explica: "*Users can use any number of addresses and their activity using one set of addresses is not inherently tied to their activity using another set, or to their real-world identity.*"

C. Escalabilidade

Para que haja um esforço computacional baixo o sistema blockchain utiliza uma árvore de Merkle, que pode ser vista na Figura 4. Essa estrutura se comporta como uma árvore hash em que sua raiz, que está contida no cabeçalho de um bloco(*block header*) da cadeia do sistema, é um hash, e a partir deste, é possível chegar até as suas folhas compostas de transações.

Com esse tipo de árvore é possível que as transações sejam acessadas sem que haja uma cópia completa de todas elas, pois, com a raiz oriunda do processo de mineração, mesmo que os ramos da árvore sejam transportados por meios não confiáveis e alterados, essas adulterações seriam detectados por possuir valor de hash diferente[7, pp. 2087].

Após uma transação ser armazenada em um bloco, é possível que outras gastas anteriormente a ela, possam ser excluídas, eliminando ramos da árvore, e assim, compactando toda cadeia e diminuindo o gasto de recursos em termos de rede e armazenamento[4, pp.4].

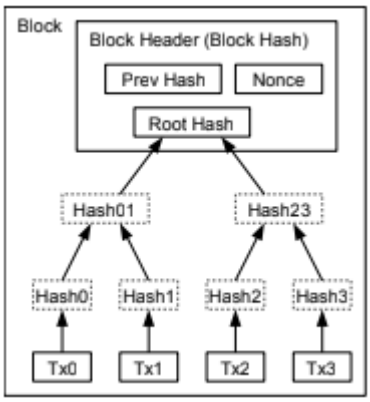

Figura 4 [4, pp.4]

Outra maneira de aumentar a escalabilidade é o envio de dados de transações a partir de solicitações das mesmas. Isso ocorre quando um nó, antes de enviar esses dados aos nós vizinhos, envia uma mensagem de inventário, contendo uma lista de hash das transações, que indica que esse mesmo nó possui mais informações sobre as transações notificadas. Assim, os nós que desconhecem esses dados reais e os desejarem solicitam ao remetente da mensagem, e com isso, eles são enviados aos mesmos. Esse processo evita transmissões desnecessárias, e consequentemente reduz a carga da rede[7, pp. 2100].

Também pode ser acrescentado, que no algoritmo de consenso PoW, os participantes não necessitam de autenticação para entrar na rede, fazendo com que a mesma tenha facilidade de crescimento em relação ao suporte a milhares de nós[22, pp. 7].

D. Concorrência

O caso de mais de um nó encontrar o valor de *hash* é bem passível de se ocorrer nas blockchains em geral, e isto pode acabar gerando um problema de partes diferentes da rede estarem trabalhando em cadeias de bloco diferentes, podendo ser causado pela distância entre os nós, onde alguns destes recebem blocos válidos de mineiros diferentes, causando uma ramificação da

cadeia de blocos original em diferentes *branches* (ramos).

O Bitcoin e as demais aplicações baseadas em Blockchain lidam com este problema, como já foi abordado no tópico 2, com a ideia de aceitar sempre a maior cadeia de blocos válidos como a correta[7, pp. 2084]. Portanto, em algum momento algum dos *branches*, provavelmente aquele que possui mais poder computacional, irá encontrar e incluir um bloco mais rápido que os demais, forçando-os a aceitar este *branch* que os ultrapassou como cadeia principal, encerrando a concorrência.

E. Tolerância a falhas

Em um sistema é muito importante que exista a tolerância a falhas para seu bom funcionamento. O blockchain garante que cada nó contenha um registro completo de todas as transações que já foram feitas[24, 13%], desenvolvendo réplicas dessas informações que ficam armazenadas em cada participante da rede, garantindo redundância, e com isso, se um nó dessa rede falhar, a mesma, não sofrerá grandes consequências[7, pp. 2084], sendo portanto, robusta.

Com isso, foi possível também, resolver um problema de sistemas distribuídos muito comum, o problema dos generais bizantinos[7, pp. 2084]. Se existir um ataque sybil, esse será detectado e evitado, pois toda a rede verifica a legitimidade das transações, onde cada participante realiza um trabalho para provar a autenticidade da mesma dependendo do poder computacional e não do número de identidades que, nesse caso, podem ser falsas (como visto na seção anterior). Assim, ataques maliciosos para gerar erros na rede e falsas identidades podem ser evitados[7, pp. 2086][22, pp. 4].

Além do blockchain ser usado amplamente pelas criptomoedas, sua aplicação pode se estender por diversas áreas, assim como para efetuar uma eleição.

Para que isso seja possível, carteiras digitais seriam criadas para cada candidato e os cidadãos receberiam um conjunto de chaves, sendo ele para sua identificação e para fazer a uma relação ao seu candidato preferido, onde, cada eleitor enviaria uma moeda, que indicaria o voto, para a carteira escolhida. O próprio sistema do blockchain, registra, válida e faz a contagem de votos, analisando cada carteira, e assim, ditando o eleito como o dono da certeira que recebeu mais votos. Os votos são mandados para diversos servidores e o resultado pode ter suas informações mostradas publicamente.

Esse processo garantiria a integridade dos dados, combatendo fraudes e erros, além de, o voto poder ser registrado a partir de qualquer lugar do mundo no *ledger* distribuído, tornando assim esse sistema melhor do que o atual[24, 6%,13%].

V. ESTADO DA ARTE

O Bitcoin foi a primeira criptomoeda que revolucionou o campo de moedas digitais por causa das inovadoras características vistas no decorrer do artigo, mas atualmente existem mais de mil criptomoedas diferentes, onde poucas conseguem se consolidar no mercado. As que conseguem se destacar são as moedas que resolveram de melhor forma algum problema, e essas serão abordadas adiante, onde será feito um comparativo entre as mesmas, destacando suas peculiaridades.

Uma forma simples de comparar as moedas digitais atuais é acessando sites de *Market Cap* (sigla de *Market Capitalization)* de criptomoedas, o qual utiliza da mesma ideia que é usada para a bolsas de valores, que consiste em uma forma classificar as criptomoedas através da multiplicação de duas variáveis. O preço da equação é obtido através de uma média ponderada do volume dos preços da moeda que o cálculo está

sendo realizado, de diferentes mercados de criptomoedas, e o fornecimento circulante é obtido por uma aproximação da quantidade de moedas que estão sendo circuladas no mercado e as que estão sendo utilizadas pelo público[27][28].

O funcionamento do market cap para criptomoedas tem gerado algumas dúvidas sobre sua eficiência, isso ocorre porque, geralmente a informação da variável de fornecimento circulante do cálculo do market cap não é fácil de ser encontrada. Um exemplo disso foi o que ocorreu com a criptomoeda Auroracoin, que em 2004 tinha um valor de market cap de um bilhão de dólares, mas sendo grande que parte desse valor não estava realmente em circulação mas sim trancando e indisponível para o comércio, pois ainda não tinham sido lançados para o público, fazendo com que o real valor do market cap girasse em torno de 10 milhões de dólares[28].

Uma forma mais eficiente de equiparar as moedas é entendendo como elas funcionam, fazendo uma comparação de suas arquiteturas para então descobrir os pontos fortes e fracos de cada uma. Nessa etapa, o artigo irá focar apenas naquelas criptomoedas que inovaram e conseguiram solucionar problemas antes já detectados em outras.

A) Scrypt

Uma das primeiras criptomoedas após o Bitcoin que conseguiu se destacar foi o Litecoin, desenvolvido por Charles Lee em outubro de 2011, que tinha como objetivo apresentar uma alternativa que possuía um processo de mineração mais leve e mais rápido, tentando ter as suas transações confirmadas em velocidade maior que o Bitcoin, e ainda, utilizar esta rede já consolidada como base de suas transações. Para isso, ele utiliza do algoritmo de hash Scrypt para o *proof-of-work,* que é focado no uso intensivo da memória ao invés do algoritmo SHA-256, utilizado pelo Bitcoin, que tem seu foco no uso intensivo do processador[29, pp. 525- 527].

O Scrypt inicialmente foi projetado para o hashing de senhas, de maneira a dificultar que elas sejam encontradas através de uma técnica de força bruta, e para isso usa da função hash memory-hard, que para ela possa ser computada é necessário um alto consumo de memória. A função hash memory-hard foi definida por seu criador Colin Percival, no artigo Stronger Key Derivation via Sequential Memory-Hard Functions, como sendo um “*algorithm which asymptotically uses almost as many memory locations as it uses operations; it can also be thought of as an algorithm which comes close to using the most memory possible for a given number of operations*” [30, pp. 219][31, pp. 3].

O alto consumo de memória no algoritmo é utilizado da seguinte forma: uma grande quantidade de bits pseudo-aleatórios é mantida na memória e uma chave é derivada disso de uma maneira pseudo-aleatória. O algoritmo é baseado em um fenômeno chamado Time-Memory Tradeoff(TMTO). Se os requisitos de memória não forem seguidos, resultará em um aumento no custo computacional, o que em outras palavras quer dizer que, o TMTO encurta o tempo de execução de um programa caso mais memória for dada a ele.

Essa desvantagem torna inviável para um invasor obter mais memória, porque é custoso e difícil de implementar em hardware personalizado, mas se o invasor preferir não aumentar a memória, isso resulta em um algoritmo mais lento devido aos altos requisitos de processamento [21, pp. 182].

O Scrypt pode ser utilizado com o mesmo quebra-cabeça da mineração do Bitcoin, mantendo a mesma taxa de dificuldade para encontrar o hash, porém conseguindo gerar um novo bloco em uma

velocidade maior do que as criptomoedas que utilizam o algoritmo SHA-256. Um exemplo seria a moeda Litecoin, citada acima, e a Dogecoin, que utilizam o algoritmo Scrypt e que conseguiram obter um tempo de confirmação de 2.5 minutos e 1 minuto, respectivamente, enquanto o Bitcoin geralmente gasta uma média de 10 minutos [7, pp. 2113][30, pp. 219].

O problema com o algoritmo Scrypt é que ele ainda gera uma desconfiança pelo fato de ainda não ter sido bem testado quanto o SHA-256[7, pp. 2113], que foi melhor analisado e testado pela agência nacional americana, como visto no [32].

B) Proof-of-Stake

Proof-of-Stake (PoS) surgiu como uma alternativa para o tradicional proof-of-work (PoW), utilizado pelo Bitcoin, e pela maior parte das criptomoedas que vieram após ele, como sistema de consenso. Apareceu pela primeira vez no PeerCoin em agosto de 2012 [33] com a principal proposta de eliminar o alto consumo de energia que é necessário na mineração tradicional, além de, adicionar mais segurança à rede com um novo mecanismo de consenso [21, pg. 166] [7, pp. 2114, 2115]. Seu conceito é baseado na ideia de *coin age* (idade da moeda), que também é usado pelo Bitcoin, mas apenas para priorizar transações que estão esperando a muito tempo.

A *coin age* de uma transação aumenta de acordo com o seu tempo e valor, sendo consumida (deletada), quando seu *output* é utilizado. No sistema do proof-of-stake esse valor consumido é utilizado para determinar a dificuldade do *hash*, diferente do Bitcoin que possui um *target* fixo para todos os nós. Nesse, quanto maior o *coin age* acumulado, mais fácil será encontrar o valor do *hash*, ou seja a dificuldade do *hash* é inversamente proporcional a *coin age* total do bloco [33, pp. 1, 3].

As recompensas em moedas no sistema do *proof-of-stake* também são baseadas no mesmo método, e a primeira transação de um bloco é chamada de *coinstake*, inspirada no *coinbase* do Bitcoin. Nela o dono do bloco com as transações, paga a si mesmo com sua própria reserva da moeda, consumindo o seu *coin age* que estava acumulado e com isto ganhando o privilégio de receber um valor derivado deste em moedas, se conseguir gerar um novo bloco para a rede [33, pp. 2, 3] [7, pp. 2114].

A mineração nesta nova abordagem reduz drasticamente o consumo de energia, por limitar sua busca em um espaço finito de tentativas pelo *hash* adequado, diferente do PoW que é virtualmente infinito. Explicando melhor, os dados utilizados na encriptação do *hash* são, com exceção do *timestamp*, estáticos, e portanto, não há um *nonce* a ser incrementado, ao invés disso a cada vez que *timestamp* muda, os mineiros têm a chance de encontrar a resposta, que terá diferentes dificuldades para cada um (dependendo da *coin age* consumida). A consequência disto é que o poder computacional se torna muito menos relevante para encontrar o *hash*, pois não é possível utilizá-lo para aumentar as chances de encontrar a solução, e portanto a quantia de *coin age* acumulada que irá, provavelmente, determinar o vencedor. Esta diferença torna a recompensa deste sistema ainda mais atraente pelo menor gasto elétrico em decorrência de não existir esta busca exaustiva e custosa financeiramente pela solução do quebra-cabeças como ocorre no PoW [33, pp. 3] [7, pp. 2114].

O mecanismo de consenso no *proof-of-stake* inova ao se basear na quantia de *coin age* que foi consumida, dessa forma cada transação do bloco contribuí com o seu *coin age* consumido para a pontuação total do bloco e aquele com o maior total de “moedas consumidas” é escolhido para fazer parte da cadeia principal. A ideia por trás desta mudança é que o custo de se controlar

uma quantia significante de moedas para realizar um ataque é maior do que o custo necessário para ter 51% do poder computacional da rede, além do que, se algum usuário possuir uma quantidade tão grande de moedas, provavelmente este seria um dos mais prejudicados com uma eventual falência da moeda. Portanto, esse sistema pode oferecer mais segurança para a rede do que o *proof-of-work* [33, pp. 3] [7, pp. 2115].

O problema deste sistema é a falta de penalidade para nós que trabalharem em mais de um *branch* da blockchain ao mesmo tempo, com o intuito de em algum momento alcançar o ramo principal. Em outras palavras, os nós podem votar em mais de uma cadeia principal, gerando o chamado "*nothing at stake problem*", que enquanto no PoW trabalhar em mais de uma cadeia é muito custoso, pois divide os recursos computacionais e portanto diminui a chance de resolver o quebra-cabeça primeiro, o esquema de PoS, com o intuito de diminuir o gasto elétrico, abaixa o custo computacional da resolução do quebra-cabeça, e portanto, permite tais ramificações, que podem ou não serem maliciosas, mas não são desejadas por tornarem o consenso da rede muito lento [34, pp. 1, 3].

C) Ethereum

O Bitcoin, como já visto, possui uma linguagem de script eficiente mas muito limitada, onde permite apenas operações básicas e necessárias para garantir que as transações de pagamento sejam realizadas, diferente do Ethereum, criada por Vitalik Buterin e lançada em 30 de julho de 2015, que propõe uma plataforma baseada em uma rede Blockchain que utiliza de uma linguagem de programação Turing Completa para a execução de smart contracts (contratos inteligentes) , que será explicado a seguir, e que permite que o desenvolvimento de diversas aplicações descentralizadas de alta confiabilidade, e não apenas, transações econômicas. [35][21, pp. 210][30, pp. 285]

Como pode ser visto no Ethereum *white paper* [36, pp.13], "*The intent of Ethereum is to merge together and improve upon the concepts of scripting, altcoins and on-chain meta-protocols, and allow developers to create arbitrary consensus-based applications that have the scalability, standardization, feature-completeness, ease of development and interoperability offered by these different paradigms all at the same time. Ethereum does this by building what is essentially the ultimate abstract foundational layer: a blockchain with a built-in Turing-complete programming language, allowing anyone to write smart contracts and decentralized applications where they can create their own arbitrary rules for ownership, transaction formats and state transition functions.*"

Os smart contracts que foram inicialmente teorizados por Nick Szabo na década de 90, como sendo "*a computerized transaction protocol that executes the terms of a contract. The general objectives are to satisfy common contractual conditions (such as payment terms, liens, confidentiality, and even enforcement), minimize exceptions both malicious and accidental, and minimize the need for trusted intermediaries. Related economic goals include lowering fraud loss, arbitrations and enforcement costs, and other transaction costs.*" [21, pp. 198], ou seja, no caso do Ethereum são programas de computador seguros que representam contratos que devem funcionar exatamente como programados, sem qualquer possibilidade de tempo de inatividade, censura, fraude ou interferência de terceiros [22, pp.29-30] [21, pp. 198].

Os smart contracts podem ser criados por qualquer usuário da rede fazendo-se o upload do código de programa do seu contrato em uma transação especial. O

código deve ser escrito em uma linguagem de bytecode baseada em pilha e de baixo nível, conhecida como EVM (Ethereum Virtual Machine) code, que é executado pela própria máquina virtual do Ethereum. Uma vez carregado o contrato passa a fazer parte da Blockchain, e pode ter seu próprio saldo de fundos, ou até, outros usuários podem fazer chamadas de procedimento através de qualquer API que o programa exponha, assim como, pode enviar e receber dinheiro [36, pp.17][30, pp. 286-287].

O Ethereum utilizava inicialmente o proof of work como o Bitcoin mas na sua última versão a *Serenity*, terá um novo algoritmo chamado Casper, no qual, será feita a mudança para proof of stake, explicado acima, mas com uma diferença, onde, ele tenta resolver o problema "*nothing at stake",* que se algum validador estiver fazendo algo que o Casper considere inválido, o seu depósito inicial necessário no proof of stake será confiscado e o mesmo banido de participar do processo de consenso. [21, pp. 248-249] [37]

O Ethereum, por ter uma linguagem Turing completa, possui uma rede muito mais complexa, o que se gera um problema de escalabilidade, pois quanto mais a rede crescer mais complexa ela fica, reduzindo assim uma das características fundamentais de um sistema distribuído.

Outro problema ocasionado por essa característica, é o fato de dar muita liberdade aos usuários, sendo esse, um dos motivos pelo qual o Bitcoin foi propositalmente feito Turing incompleto, limitando o leque de formas de que as transações pudessem ser feitas. Essa liberdade pode ocasionar a criação de smart contracts com bugs e falhas de seguranças que ficam visíveis para toda a rede, que pode tentar tirar proveito desta falha, e essas, não podem ser concertadas rapidamente, pois uma das características dos smarts contracts é fato de que depois que um contrato é criado ele não pode ser facilmente alterado [22, pp. 30][38].

D) Proof of space

A ideia de proof of space (PoSpace) foi introduzida por Dziembowski et al. [39] e seu uso foi proposto em 2015 no *whitepaper* da criptomoeda Spacecoin [34] que visa, assim como o PoS, uma alternativa ao PoW do Bitcoin, onde desta vez, no lugar de poder computacional, esse é baseado em espaço em disco [34, pp. 1].

Esta abordagem pede que um nó prove que reservou uma certa quantidade de espaço em disco, assim como dito no artigo Proofs of Space [39, pp. 2]: "*PoSpace are partially motivated by the observation that users often have significant amounts of free disk anyway, and in this case using a PoS is essentially for free*".

Neste sistema em vez de um invasor necessitar de mais poder computacional que a rede, este precisaria de mais espaço em disco dedicado, sendo portanto uma solução mais sustentável pelo baixo gasto energético, pois uma vez que um minerador tenha dedicado o espaço em disco requerido, o processo de mineração é barato. O Spacecoin sugere que blocos sejam adicionados em um período determinado de tempo, bastando uma pequena série de verificações para que um nó possa ter conhecimento se é o vencedor, evitando a corrida de competição que o PoW e processos semelhantes estimulam, e completa [34, pp. 6]: "*miners only have to execute a proof once every minute, but apart from that can use their resources (except the space dedicated for mining) in a useful way.*" [34, pp. 1].

Esse mecanismo funciona com um protocolo de duas fases, uma de inicialização e uma de execução, onde um verificador envia ou compartilha de alguma forma um arquivo, que pode ser aleatório, de tamanho

N para o usuário que deseja participar da mineração. O verificador guarda uma pequena parte deste arquivo (*short commitment*), para na fase seguinte, de execução, possa usá-la para propor um desafio ao usuário, onde este deve provar que ainda mantém o arquivo completo guardado. O verificador portanto solicita partes aleatórias do arquivo, que foram guardadas pelo minerador, o nó deve enviá-las para a validação, e realizar esta tarefa dentro de um tempo esperado, em função do tamanho N (Ex: Θ(N)) do arquivo (mostrando que precisou fazer uma busca no mesmo) [34, pp. 4, 5] [40, pp. 2]. Este protocolo possui alguns problemas, como a possibilidade de reúso do mesmo espaço alocado, com algumas poucas mudanças, para realizar mais de uma prova, além de cair no mesmo problema que existe no proof of stake clássico (*nothing at stake*), não ter penalidade para um minerador que trabalhe em mais de uma cadeia ao mesmo tempo [34, pp. 1, 9] [40, pp. 2].

O Spacecoin aplica esta ideia as criptomoedas, mantendo os incentivos do Bitcoin de transações e blocos, criando um tipo especial de transação que um novo minerador deve fazer, gravando na blockchain uma associação da sua chave pública com este *short commitment*, chamado de *space commitment*, gerado por ele na fase de inicialização, para um bloco dedicado de tamanho N[34, pp. 7, 8, 12]. Portanto, o processo de mineração começa com uma *chalenge* C, que pode vir de várias fontes. Os autores de [34, pp. 11, 16] sugerem que derive do número do bloco atual i - dist (dist > 1), e esse é utilizado com a chave pública, o *space commitment*, e outras informações do bloco para criar um *hash*. O nó vencedor (que encontrou o melhor *hash*) deverá ser escolhido de forma que a sua chance de vitória seja proporcional a sua fração de espaço em disco dedicado, em relação ao total dos demais participantes[34, pp. 8].

A transação de *space commitment* impede que um nó grave um novo espaço de memória com a mesma chave pública, evitando, portanto, o problema do reuso de memória, mas ainda assim persistindo o problema de trabalho em mais de uma *branch*. Para este, a solução adotada é uma *punishment transaction* para desmotivar nós, roubando a recompensa ganha[34, pp. 10, 12, 15]. Outro problema que ocorre com o proof of space, e que não é solucionado pelo Spacecoin, é que este apenas assume que os nós irão alocar o espaço honestamente, mas um nó pode salvar apenas a parte do *commitment* criado na inicialização e apenas refazer este processo durante a fase de execução, ou seja, trocando espaço em disco por processamento, retornando, dessa forma, a um proof of work em termos de gasto elétrico[34, pp. 17][40, pp. 2, 3].

VI. CONCLUSÃO

Em termos gerais, o blockchain introduz ao mundo dos sistemas distribuídos uma característica nunca antes vista, onde, uma vez que alguma informação é gravada na blockchain é quase impossível conseguir removê-la ou alterá-la, e isso beneficia aplicações que antes não tinham grande avanço, como exemplo as criptomoedas, que retomaram sua pesquisa após o Bitcoin, como foi visto neste artigo. Além de ter uma boa abordagem, o principal problema que este apresenta é o enorme gasto energético demandado, e a partir dele novos mecanismos começam a serem estudados, assim gerando novos termos como proof of stake, proof of space, Scrypt do Litecoin, Casper do Ethereum e muitos outros que não foram abordados por este artigo, assim como também, melhorando características como tolerância a falhas, privacidade e

transparência, que já eram notadas desde o seu início.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] V. Kostakis, C. Giotitsas, “The (A)Political Economy of Bitcoin”, *TRIPLEC*, vol. 12, pp. 431-440, Junho 2014.

[2] B. Schröder, (2015) “O Que é Moeda?” on Academia Liberalismo Econômico. [Online]. Available: https://aleconomico.org.br/o-que-e-moeda

[3] E. N. da Silva, S. S. P. Júnior, “Sistema Financeiro e Crescimento Econômico: uma Aplicação de Regressão Quantílica”, *Economia Aplicada*, vol. 10, no. 3, p. 425-442, Setembro 2006.

[4] S. Nakamoto, (2008) “Bitcoin: A Peer-to-Peer Electronic Cash System,”, [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[5] D. Kraft, “Difficulty Control for Blockchain-Based Consensus Systems”, *Peer-to-Peer Networking and Applications*, vol. 9, pp. 397–413, Abril 2015.

[6] S. Meiklejohn, M.i Pomarole, G. Jordan, K.l Levchenko, D. McCoy, G. M. Voelker, S. Savage, “A Fistful of Bitcoins: Characterizing Payments among Men with No Names”, *Communications of the ACM*, vol.. 59, No. 4, pp. 86-93, abril 2016.

[7] F. Tschorsch, B. Scheuermann, “Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies”, *IEEE Communications Surveys & Tutorials Journal*, vol.18, no. 3, pp. 2084-2123, Agosto 2016.

[8] P. G. Dwyer, “The Economics of Bitcoin and Similar Private Digital Currencies”, *Journal of Financial Stability*, Vol. 17, pp. 81-91, Abril 2015.

[9] Banking on Bitcoin. Direção: Christopher Cannucciari. Produção: Christopher Cannucciari, David Guy Levy. EUA, 2016. Documentário (90 min.).

[10] D. Chaum, ”Achieving Electronic Privacy”, Scientific American, Vol. 267, No. 2, pp. 96-101, Agosto 1992.

[11] N. Asokan, P. A. Janson, M. Steiner, M. Waidner, “The State of the Art in Electronic Payment Systems”, *IEEE Computer Society*, vol. 30, no. 9, pp. 28–35, Setembro. 1997.

[12] K. A. Strassel. (1996) “Deutsche Bank to Test 'E-Cash' With DigiCash in Pilot Project” on The Wall Street Journal. [Online]. Available:https://www.wsj.com/articles/SB831416067295410500

[13] L. C. Bresser-Pereira, “Crise e recuperação da confiança”, *Revista de Economia Política*, vol. 29, no. 1, pp. 133-149, Janeiro 2009.

[14] W. Dai. (1998). B-Money [Online]. Available: http://www.weidai.com/bmoney.txt

[15] N. Szabo. (2005). Bit Gold [Online]. Available: http://unenumerated.blogspot.de/2005/12/bit-gold.html

[16] V. Vishnumurthy, S. Chandrakumar, E. G. Sirer, “KARMA: A Secure Economic Framework for Peer-to-Peer Resource Sharing”, Proceedings of the 1st Workshop on Economics of Peer-to-Peer Systems, Junho 2003.

[17] H. Finney. (2004). Rpow [Online]. Available: http://cryptome.org/rpow.html

[18] B. Weber, “Bitcoin and the Legitimacy Crisis of Money”, *Cambridge Journal of Economics*, vol. 40, pp. 17–41, Janeiro 2016.

[19] A. Yelowitz, M. Wilson, “Characteristics of Bitcoin Users: An Analysis of Google Search Data”, *Applied Economics Letters*, vol. 22, no. 13, pp. 1030–1036 , Janeiro 2015.

[20] X. Li, C. A. Wang, “The Technology and Economic Determinants of Cryptocurrency Exchange Rates: The Case of Bitcoin”,

Decision Support Systems, vol. 95, pp. 49-60, Março 2017. A1 (CC)

[21] I. Bashir, *Mastering Blockchain,* Birmingham, UK: Packt Publishing, Março 2017

[22] M. Conti, S. Kumar E, C. Lal and S. Ruj, "A Survey on Security and Privacy Issues of Bitcoin," *IEEE Communications Surveys and Tutorials,* Maio 2018

[23] B. A. Forouzan e F. Mosharraf, *Redes de Computadores: Uma Abordagem Top-Down,* Porto Alegre, BR: Amgh Editora, 2013

[24] D. Tapscott, A. Tapscott, *Blockchain Revolution,* New York, USA: Penguin Random House, 2016

[25] S. Underwood, "Blockchain Beyond Bitcoin", *Communications of the ACM,* vol. 59, no. 11, pp. 15-17, Novembro 2016

[26] M. van Steen e A. S. Tanenbaum, *Distributed Systems,* 3.01nd ed., 2017

[27] CoinMarketCap[Online]. Available:https://coinmarketcap.com/faq

[28] K. Torpey, (2017) Comparing Bitcoin and Other Cryptocurrencies by 'Market Cap' Can Be Very Misleading on Forbes [Online], Available:https://www.forbes.com/sites/ktorpey/2017/12/29/comparing-bitcoin-and-other-cryptocurrencies-by-market-cap-can-be-very-misleading/#373282682509

[29] G. Silva e C. Rodrigues, "Mineração individual de bitcoins e litecoins no mundo", Conference: XVI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais — SBSeg 2016, pp. 524-533, Novembro 2016

[30] A. Narayanan, J. Bonneau, E. Felten, A, Miller, S. Goldfeder, *Bitcoin and Cryptocurrency Technologies,* New Jersey, USA: Princeton University Press , 2016.

[31] C. Percival, "Stronger key derivation via sequential memory-hard functions", pp. 1-16 ,Janeiro 2009

[32] D. Eastlake III and T. Hansen. (2011). US Secure Hash Algorithms (SHA and SHA-Based HMAC and HKDF), RFC 6234 (Informational), Internet Engineering Task Force [Online]. Available: http://www.ietf.org/rfc/rfc6234.txt

[33] S. King e S. Nadal, "PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake", pp. 1-6, Agosto 2012

[34] S. Park, K. Pietrzak, J. Alwen, G. Fuchsbauer e P. Gazi, "Spacecoin: A Cryptocurrency Based on Proofs of Space", pp. 1-29, Junho 2015

[35] History of Ethereum on Ethereum Homestead Documentation [Online]. Available: http://ethdocs.org/en/latest/introduction/history-of-ethereum.html

[36] V. Buterin, "Ethereum White Paper: A Next-Generation Smart Contract and Decentralized Application Platform", pp.1-36

[37] V. Zamfir. (2015) Introducing Casper "the Friendly Ghost" on Ethereum Blog [Online]. Available: https://blog.ethereum.org/2015/08/01/introducing-casper-friendly-ghost/

[38] M. E. Peck. (2016) Ethereum's $150-Million Blockchain-Powered Fund Opens Just as Researchers Call For a Halt on IEEE Spectrum [Online].Available:https://spectrum.ieee.org/tech-talk/computing/networks/ethereums-150-million-dollar-dao-opens-for-business-just-as-researchers-call-for-a-moratorium

[39] S. Dziembowski, S. Faust, V. Kolmogorov e K. Pietrzak, "Proofs of Space", pp. 1-31, 2015

[40] T. Moran e I. Orlov "Proofs of Space-Time and Rational Proofs of Storage", pp.1-22, 2016