Sistema Bitcoin: uma análise da segurança das transações

Carlo Kleber da Silva Rodrigues[1]

[1]Faculdade de Tecnologias e Ciências Sociais Aplicadas – FATECS
Centro Universitário de Brasília (UniCEUB)
Brasília – DF – Brasil

carlokleber@gmail.com

Abstract. *This paper is mainly focused on analysing the level of security associated to Bitcoin system transactions. To this end, a detailed explanation of each activity taking part of the execution of a system transaction is firstly presented. Then, through mathematical modeling, experiments to assess the level of security of a system transaction are performed. The final results allow us to conjecture about the technological strength of the Bitcoin system. Lastly, avenues for further research are included at the end of this paper as well.*

Resumo. *Este artigo concentra-se principalmente na análise do nível de segurança associado às transações do sistema Bitcoin. Para isso, é inicialmente apresentada uma explicação detalhada de cada atividade que integra a execução de uma transação no sistema. Em seguida, através de modelagem matemática, são realizados experimentos para avaliar o nível de segurança de uma transação. Os resultados finais permitem conjecturar sobre a significativa robustez tecnológica do sistema Bitcoin. Por último, sugestões para futuras pesquisas, dentro deste tema de pesquisa, estão incluídas no final deste documento.*

1. Introdução

Os sistemas de pagamentos eletrônicos tradicionais dependem da existência de instituições financeiras fidedignas, i.e., uma terceira parte formal e confiável, para executar e validar as transações realizadas. Essa dependência se dá pelo fato de as transações não serem garantidas como completamente não reversíveis, devido principalmente à existência de fraudes cada vez mais sofisticadas. Surge assim a necessidade da mediação de disputas entre as partes envolvidas, gerando incertezas e encarecendo significativamente o processo como um todo.

A implementação pura e simples de assinaturas digitais [Kurose e Ross 2013; Tanenbaum e Wetherall 2010] é apenas uma solução parcial para o cenário descrito anteriormente. Apesar de essas permitirem que uma transação possa ser feita diretamente entre duas partes interessadas, com a devida e necessária comprovação de autoria, propriedade e/ou concordância de conteúdo, ainda se exige a intervenção de uma terceira parte para evitar o chamado gasto duplicado, ou seja, que o mesmo recurso financeiro seja empregado em mais de uma transação.

Ante o exposto e levando-se em conta o crescente comércio eletrônico, devido à popularidade já alcançada pela Internet, a concepção de um sistema eletrônico de transações financeiras baseado em uma robusta prova criptográfica, que elimine a

necessidade de validação por uma terceira parte e garanta a irreversibilidade da transação e, ainda, impeça o gasto duplicado, torna-se extremamente atrativa. Essa concepção é implementada pelo sistema de pagamento eletrônico *Bitcoin* [Nakamoto 2008].

A segurança da realização de uma transação no sistema *Bitcoin* não se respalda exclusivamente na robustez do algoritmo criptográfico utilizado, ou seja, na impossibilidade da quebra de uma chave criptográfica. A segurança desse sistema considera ainda o tempo de registro da transação então realizada. Nesse sentido, quanto mais antigo é o registro da transação no sistema, maior é a segurança e, consequentemente, menos provável a sua violabilidade.

Dentro deste contexto, este artigo tem o objetivo principal de mensurar o nível de segurança de transações realizadas no sistema *Bitcoin*, considerando especialmente o tempo de registro dessas transações no sistema. Para tanto, é apresentada inicialmente uma explicação detalhada de cada atividade que integra a execução de uma transação no sistema. Em seguida, por meio da modelagem matemática de um cenário em que um suposto fraudador busca introduzir uma transação falsa no sistema, são realizados experimentos para avaliação do nível de segurança. Os resultados finais permitem conjecturar sobre a força tecnológica do sistema *Bitcoin*.

Ante o objetivo a ser alcançado, este artigo apresenta, como sua maior contribuição para a literatura especializada, a disponibilização de um único texto que consegue dar ao leitor um entendimento pleno de como ocorre uma transação no sistema *Bitcoin*, bem como uma análise científica analítica da segurança das transações realizadas nesse sistema.

O restante deste texto está organziado como descrito a seguir. A Seção 2 tem os fundamentos para o entendimento da pesquisa aqui desenvolvida. Trabalhos relacionados estão na Seção 3. A Seção 4 explica o sistema *Bitcoin*. A Seção 5 conduz uma análise de performance focada na segurança das transações. Finalmente, conclusões finais e trabalhos futuros aparecem na Seção 6.

2. Fundamentos

2.1. Criptomoedas

A criptomoeda é uma moeda virtual [Dwyer 2011] que utiliza criptografia, impedindo primordialmente que esta seja falsificada. Outro aspecto decorrente do emprego da criptografia é o anonimato oferecido aos clientes das transações. Estes clientes são vistos apenas como números criptografados, i.e., por meio de suas assinaturas digitais, com fins exclusivos de confirmação de autoria, propriedade e/ou concordância, sem a identificação nominal. Isso preserva a inteira privacidade dos clientes.

A denominação específica de cada criptomoeda advém da denominação do sistema de transações a qual ela pertence. De forma geral, a denominação do sistema tem a letra inicial maiúscula, enquanto a criptomoeda relacionada tem a mesma denominação com a letra inicial minúscula. Por exemplo, o sistema *Bitcoin* utiliza a criptomoeda *bitcoin* (abreviada por BTC), que é considerada a primeira criptomoeda implementada e hoje é a mais utilizada.

A criptomoeda não é emitida por uma instituição financeira, i.e., uma autoridade central como um banco oficial do governo de um país. Essa condição faz com que a criptomoeda esteja imune, teoricamente, a interferências por parte do governo, impedindo-o de estabelecer custos, i.e., taxas e/ou semelhantes, para a execução das transações.

Perceba que estar livre da interferência do governo e ter garantido o anonimato nas transações executadas têm o seu lado negativo também. Essa situação torna a criptomoeda propícia para o seu emprego em transações ilícitas como, por exemplo, lavagem de dinheiro, evasão fiscal e financiamento de ações terroristas.

Ainda, como as cotações da criptomoeda flutuam em função da lei da oferta e da procura, observa-se a existência de um mercado significativamente instável [Dyhrberg 2016]. Por exemplo, em 2013, a criptomoeda *bitcoin* teve uma valorização de 6.000% em apenas um ano, alcançando a marca de USD 1.250,00 por *bitcoin*. Mas, no ano seguinte, despencou e perdeu 2/3 do seu valor. Atualmente, um *bitcoin* está cotado em cerca de USD 379,00 [COINDESK 2016].

Outro aspecto negativo, desta vez considerando-se a implementação tecnológica do sistema de criptomoedas em si, é o fato de que, como se trata de um sistema virtual e sem um repositório de dados logicamente centralizado, o saldo dos usuários do sistema pode ser simplesmente apagado em virtude de uma pane computacional generalizada, caso não exista um robusto e consistente *back-up* de todos os dados do sistema [WEUSECOINS 2016].

Além do *Bitcoin*, alvo da pesquisa deste trabalho, existem outros sistemas de criptomoedas como, por exemplo, *Litecoin*, *Namecoin* e *Peercoin* [WEUSECOINS 2016]. Cada um deles, além das considerações gerais anteriores, possuem características próprias e exclusivas que, a depender do cenário da transação financeira, podem favorecer a escolha de uma ou outra.

2.2. Sistemas criptográficos

De forma simples, sistemas criptográficos [Kurose e Ross 2013; Tanenbaum e Wetherall 2010] permitem que um emissor *E* proteja a mensagem *m* a ser transmitida de tal sorte que um desconhecido *D* não consiga decifrar a mensagem *m*, caso ela venha a ser interceptada durante o processo de transmissão. Teoricamente, apenas o receptor *R* é capaz de decifrar a mensagem *m* enviada.

A proteção mencionada consiste em aplicar um algoritmo de criptografia que codifica a mensagem *m* de texto em claro em um texto cifrado antes de transmitir. Os algoritmos de criptografia dos sistemas mais modernos são de domínio público. O que garante a indecifrabilidade da mensagem *m* é o conceito do emprego de chaves utilizado pelo algoritmo de criptografia, conforme explicado a seguir.

Na transmissão, a entrada do algoritmo de criptografia é a mensagem *m*, em texto em claro, e a chave *KE* do emissor. Essa chave consiste em uma cadeia de caracteres alfanuméricos. A saída do algoritmo de criptografia é o texto cifrado $KE(m)$, que é transmitido por meio da rede. Nesse caso, o algoritmo de criptografia é dito algoritmo de encriptação. Na recepção, quando a mensagem em texto cifrado chegar ao destino, um processo inverso é realizado. O receptor utiliza a sua chave *KR* e o texto

cifrado $KE(m)$ como entrada do algoritmo de criptografia, o qual fornece, como saída, a mensagem de texto em claro novamente: $KR(KE(m)) = m$. Nesse caso, o algoritmo de criptografia é dito algoritmo de decriptação (ou desencriptação).

De maneira geral, a ação de um algoritmo de criptografia, considerando o processo de encriptação, consiste em substituir uma cadeia de caracteres do texto em claro por outra cadeia diferente de caracteres, resultante de uma operação matemática e correspondendo ao texto cifrado. Já a ação referente ao processo de decriptação consiste em realizar exatamente o oposto: identificar a cadeia de caracteres original que corresponde a cadeia que é recebida.

Em sistemas criptográficos de chaves simétricas, o Emissor *E* e o Receptor *R* têm chaves idênticas, ou seja, $KE = KR$. Dois conhecidos padrões para sistemas desse tipo são o *Data Encryption Standard* (*DES*) e o *Advanced Encryption Standard* (*AES*) [Kurose e Ross 2013; Tanenbaum e Wetherall 2010].

No caso dos sistemas criptográficos de chave pública, um par de chaves diferentes é utilizado: uma chave privada e uma chave pública. A chave privada é de conhecimento exclusivo do Emissor *E* ou do Receptor *R*, mas nunca de ambos. Já a chave pública é de conhecimento tanto do Emissor *E* quanto do Receptor *R*, além de todos que estão na rede. A operação desse tipo de sistema é relativamente simples, conforme resumimos a seguir.

Admita que a chave privada e a chave pública do Receptor *R* são *KP* e *KB*, respectivamente. Para transmitir uma mensagem para o Receptor *R*, o Emissor *E* então utiliza a chave pública *KB* para cifrar a mensagem *m*, obtendo, na saída do algoritmo, o texto cifrado $KB(m)$. No destino, o Receptor *R* precisa apenas utilizar sua chave privada *KP* para decifrar a mensagem $KP(m)$ então recebida do Emissor *E*. Em notação simbólica, o processo no Receptor *R* seria então: $KP(KB(m)) = m$. Lembre que a chave pública *KB* do Receptor *R* é conhecida por todos que estão na rede. Por definição, é importante dizer que esse sistema goza da propriedade: $KB(KP(m)) = KP(KB(m)) = m$.

Do parágrafo anterior, é possível notar a preocupação explicada a seguir. Como a chave pública *KB* do Receptor *R* é conhecida por todos que estão na rede, então qualquer Desconhecido *D* pode transmitir uma mensagem criptografada $KB(m)$ sem a necessidade de se identificar. Ou seja, não é possível saber com certeza quem transmitiu a mensagem *m* para o Receptor *R*. Para resolver essa preocupação é preciso utilizar o conceito de *assinatura digital*, discutido na próxima seção [Kurose e Ross 2013].

Dentre os padrões utilizados na implementação de sistemas criptográficos de chave pública, aquele denominado de *RSA* (Rivest, Shamir e Adleman) merece destaque por sua eficiência e ampla aceitação. A seleção da chave pública e da chave privada considera um processo matemático intricado baseado em dois números primos. Por fim, o algoritmo de criptografia utilizado (encriptação e decriptação) é baseado em operações matemáticas complexas envolvendo exponenciação e módulo [Kurose e Ross 2013; Tanenbaum e Wetherall 2010].

2.3. Assinatura digital

Em um mundo digital, deseja-se frequentemente identificar o proprietário ou autor de um documento, ou ainda deixar claro que alguém concorda com o conteúdo de um

documento. A assinatura digital [Kurose e Ross 2013; Tanenbaum e Wetherall 2010] é, pois, a técnica criptográfica para se atingir esses objetivos em um mundo digital.

A assinatura digital deve ser implementada de tal maneira que seja verificável, não falsificável e não repudiável. Isso é conseguido por meio da utilização de criptografia de chaves públicas, conforme esclarecido a seguir.

Admita que se deseja assinar digitalmente uma mensagem *m*. Admita também que a chave privada e a chave pública do Emissor *E* são *KP* e *KB*, respectivamente. O Emissor *E* então utiliza sua chave privada *KP* para realizar o processamento de *m*, por meio do algoritmo de criptografia (encriptação), resultando na saída *KP*(*m*). Nesse momento, se diz então que o Emissor *E* tem a mensagem *m* e sua assinatura digital da mensagem *KP*(*m*), as quais são ambas disponibilizadas ao Receptor *R*.

Para afirmar que a mensagem *m* é de fato assinada pelo Emissor *E*, o Receptor *R* deve proceder como explicado a seguir. O Receptor *R* utiliza a chave pública do Emissor (a saber, *KB*) para realizar o processamento de *KP*(*m*), por meio do algoritmo de criptografia (decriptação), resultando na saída *KB*(*KP*(*m*)). Esta saída é então comparada com a mensagem m para confirmar que são idênticas ou não. Caso sejam idênticas (i.e., ocorra a identidade), então de fato a mensagem *m* é assinada pelo Emissor *E*, confirmando a sua autoria ou concordância com o conteúdo; caso contrário, nada pode ser afirmado.

2.4. Função hash para assinatura digital

Muitas mensagens trocadas em rede não necessariamente precisam estar inteiramente encriptadas. Às vezes, precisa-se apenas garantir que: o Emissor *E* transmitiu a mensagem e a sua assinatura pode ser confirmada pelo Receptor *R* (i.e., autoria); a mensagem transmitida não foi modificada desde o instante em que o Emissor *E* a criou e a assinou (i.e., integridade). Com isso em mente, apresenta-se a seguir uma simplificação para o processo tradicional descrito na seção anterior com respeito a assinatura digital.

Em um primeiro momento é gerado um resumo criptográfico da mensagem através de algoritmos de função de *hash* como, por exemplo, *MD5*, *SHA-1* e *SHA-256* [Kurose e Ross 2013], que reduzem qualquer mensagem sempre a um resumo de mesmo tamanho. A este resumo criptográfico, obtido por meio de uma função de *hash*, se dá também o nome de *hash*.

Uma função de *hash* deve apresentar necessariamente as seguintes características: (i) deve ser impossível encontrar a mensagem original a partir do *hash* da mensagem; (ii) o *hash* deve parecer aleatório, mesmo que o algoritmo seja conhecido. Uma função de *hash* é dita forte se a mudança de um *bit* na mensagem original resulta em um novo *hash* totalmente diferente; (iii) deve ser impossível encontrar duas mensagens diferentes que levam a um mesmo *hash*.

Admita que *H*(*m*) é o *hash* da mensagem original *m*, o algoritmo de *hash* utilizado é conhecido pelo Emissor *E* e o pelo receptor *R*, a chave privada e a chave pública do Emissor *E* são *KP* e *KB*, respectivamente. O Emissor *E* então disponibiliza para o Receptor *R* o par de valores (*m*, *KP*(*H*(*m*)). Note que *KP*(*H*(*m*) é o *hash* assinado (criptografado) digitalmente pelo Emissor *E*, usando o conceito de assinatura privada.

Ao receber o par de valores (m, $KP(H(m))$, o Receptor R então aplica a chave pública do Emissor E para obter o *hash* da mensagem m original: $KB(KP(H(m)) = H(m)$. Em seguida, aplica o algoritmo de função *hash* conhecido ao valor de m recebido, obtendo, por exemplo, $H'(m)$. Agora, se $H(m) = H'(m)$, então o Receptor R tem confirmada a integridade e autoria da mensagem recebida; caso contrário, a mensagem não deve ser aceita.

3. Trabalhos Relacionados

A segurança, a eficiência e a longevidade do sistema *Bitcoin* têm sido alvos de estudo e análise principalmente em fóruns e *blogs* na Internet. No entanto, sob o prisma mais formal de publicações científicas e tecnológicas, a realidade é que não há muito material de estudo e análise disponível.

Também é importante novamente dizer que a segurança do sistema *Bitcoin* não está exclusivamente conferida pelo algoritmo criptográfico utilizado. A segurança reside principalmente na consideração do tempo de registro da transação no sistema. Quanto mais antigo for o registro, menos provável é a violabilidade da transação. Nesse sentido, a busca de trabalhos que analisam o algoritmo criptográfico utilizado pelo *Bitcoin* não foi considerada neste nosso trabalho; além disso, essa análise já está divulgada na literatura por tratar-se do conhecido algoritmo *SHA-256* [Kurose e Ross 2013].

Ante o exposto, esta seção busca então discorrer brevemente sobre alguns dos trabalhos científicos e tecnológicos mais importantes ou recentes da literatura que contribuem ou se relacionam, mesmo que de forma indireta, com o objetivo deste trabalho, permitindo ao leitor ter, principalmente, uma visão do estado da arte desta área de pesquisa.

O sistema *Bitcoin* foi originalmente proposto por Nakamoto (2008). O autor retrata a filosofia e o mecanismo geral de funcionamento, os conceitos técnicos aplicados e a infraestrutura de rede *peer-to-peer* (*P2P*) utilizada. Em especial, é evidenciada a necessidade do uso de um servidor de carimbo de tempo para gerar a *prova de trabalho* (ou *proof-of-work*) na ordem cronológica das transações que são realizadas.

O trabalho de Rosenfeld (2011) analisa a rentabilidade econômica da estratégia de mineração em *pools*, onde há conjuntos de mineiros (ou mineradores) trabalhando de forma colaborativa para validar uma transação realizada no sistema. O autor conclui que a vantagem desse novo paradigma de mineração ocorre em virtude, principalmente, da alta variância observada nas recompensas obtidas por meio da mineração individual, onde os mineiros trabalham isoladamente.

Luther (2015) investiga a possiblidade de as criptomoedas serem mais amplamente usadas ou se tornaram dinheiro de um nicho mais específico do mercado. O autor concluiu que, apesar de o *bitcoin* representar de fato um avanço tecnológico no processamento de pagamentos eletrônicos, é possível que as criptomoedas posteriormente concebidas venham a introduzir novos conceitos tecnológicos, redundando em sistemas mais eficientes e, assim, venham a torná-lo obsoleto e conhecido apenas como o precursor das criptomoedas.

Chávez e Rodrigues (2015) discutem formas de decidir matematicamente quando é mais vantajoso trocar de *pools* de mineiros a fim de manter a rentabilidade econômica desejada. O trabalho concluiu que a mineração considerando saltos entre *pools* é mais eficiente que a mineração considerando apenas um único *pool*. Esses mesmos autores ainda estendem o trabalho incluindo uma discussão sobre plataformas distribuídas [Chávez e Rodrigues 2016].

O trabalho de Pazmiño e Rodrigues (2015) avalia o tempo de verificação de transações e, neste contexto, propõe um esquema para a divisão da base de dados de um nó da rede *Bitcoin*, levando-se em consideração o *hardware* disponível localmente em um cliente do sistema. Os resultados finais alcançam reduções de até 71,42% no tempo de verificação de transação.

Considerando a complexidade do sistema *Bitcoin*, bem como o questionamento sobre sua adequação para a eventual substituição de um sistema financeiro monetário tradicional, Roth (2015) realiza uma análise funcional, empregando a linguagem *SysML* (*Systems Modeling Language*), com o intuito de melhor compreender a estrutura e a funcionalidade do sistema e, ainda, determinar se as expectativas e necessidades dos clientes podem ser de fato atendidas satisfatoriamente.

O trabalho de Rocha e Rodrigues (2016) tem um viés de análise baseado em metodologias de desenvolvimento de *software* e, precipuamente, apresenta a modelagem do processo de negócio do sistema *Bitcoin*. Para tanto, são empregadas técnicas de Engenharia de *Software* e de *Business Process Model and Notation* (*BPMN*). Os modelos desenvolvidos permitem mais facilmente mapear problemas críticos do sistema.

Silva e Rodrigues (2016) fazem uma análise competitiva entre os sistemas *Bitcoin* e *Litecoin*, buscando identificar qual desntre as duas seria mais rentável de se minerar considenado o modelo de mineração individual. Os resultados apontam para uma mineração mais atrativa de moedas litecoins.

Por fim, Kiayias e Panagiotakos (2016) comparam a eficiência da tecnologia de *blockchain* sob a ótica da eficiência e da segurança. Dentre os trabalhos aqui elencados, este é aquele que mais se aproxima do tema *segurança* de forma explícita. Os autores consideram uma estrutura de dados alternativa baseada em árvores para efeito de competição com aquela da *blockchain*, baseada em cadeias de blocos. Por meio principalmente de desenvolvimento analítico, o trabalho consegue apontar que, nos piores cenário de segurança estáticos, os protocolos mais modernos, como o denominado *GHOST*, têm um desempenho pior ou, na melhor das hipóteses, similar àquele da *blockchain*. Esse trabalho merece ainda destaque porque, devido à sua recente data de publicação, garante uma visão crítica do estado da arte do tema em discussão.

4. Sistema Bitcoin

A operação do sistema *Bitcoin* [BITCOIN 2016; BITCOIN SIMPLIFIED 2016] é essencialmente baseada na gerência eficiente de um *ledger* público (livro-razão) denominado de *blockchain*. Esse *ledger* contém todas as transações realizadas desde a criação do sistema, permitindo a todos os participantes verificar o histórico de cada

transação. A autenticidade de cada transação é protegida por assinaturas digitais, associadas aos *endereços bitcoin* de quem as realizou.

Qualquer participante do sistema pode realizar a validação de transações e ganhar uma recompensa por esse serviço. Essa tarefa é chamada de mineração, e aquele que a realiza é chamado de mineiros (ou minerador). Toda vez que uma transação é realizada e verificada, o *ledger* público é atualizado em todos os nós do sistema [Roth, 2015; Feld et al., 2014].

Com o intuito de melhor detalhar a operação do sistema *Bitcoin*, simplificadamente descrita anteriormente, esta seção opta por analisar todo o processo desencadeado para a execução de uma transação de pagamento por um produto (ou serviço). Para maior facilidade de exposição, as atividades constituintes desse processo são agrupadas em três distintos fluxogramas, sendo cada um deles o objeto de estudo de uma das subseções seguintes, conforme mencionado a seguir.

A Subseção 4.1 aborda as atividades realizadas pelo Cliente, aquele que realiza o pagamento referente à compra de um produto (ou serviço). A Subseção 4.2 aborda as atividades relacionadas à Rede, a qual tem a responsabilidade principal de validar o pagamento, i.e., a transação. Por fim, a Subseção 4.3 trata das atividades realizadas pelo Fornecedor, aquele que vende o produto (ou serviço) e, portanto, recebe o pagamento devido à transação realizada.

4.1. Atividades do Cliente

A partir da Figura 1, percebem-se as atividades a serem desencadeadas pelo Cliente. A primeira atividade refere-se a *Abrir Wallet*. A *wallet* é o equivalente a uma conta bancária do sistema financeiro tradicional. Ela permite que o Cliente possa receber, guardar e enviar *bitcoins* para outros participantes do sistema como, por exemplo, para o Fornecedor de um produto [BITCOIN SIMPLIFIED 2016; COINDESK 2016].

Há dois principais tipos de *wallets*: *software wallet* e *web wallet*. No primeiro tipo, o Cliente instala a *wallet* em seu próprio *hardware* (p. ex., *desktop*, *smartphone*, etc.). Neste caso, o Cliente possui total gerência relacionada a suas moedas. No segundo tipo, também chamado de *host wallet*, o Cliente hospeda sua *wallet* em uma terceira parte (p. ex., um provedor de serviços), contratada para realizar a gerência de suas moedas.

Ao abrir uma *wallet*, o Cliente passa a ter associado um *endereço bitcoin* de domínio público, como um endereço de *email* comum. Por meio desse endereço, o Cliente pode então receber transferências em *bitcoins* realizadas para ele. Semelhantemente, para transferir *bitcoins* para um participante do sistema, basta que o Cliente indique o *endereço bitcoin* desse participante. Ressalta-se que os *endereços bitcoin* são criados de forma privada, garantindo-se total anonimato de quem os cria.

Seguindo a Figura 1, a segunda atividade é *Escolher Fornecedor*. Admitindo-se que o Fornecedor tenha uma *wallet* associada a ele, a execução dessa atividade consiste em o Cliente simplesmente indicar o *endereço bitcoin* do Fornecedor.

A terceira atividade é *Submeter Transação*. Como mencionado, toda transação é registrada em um *ledger* público, denominado de *blockchain*, e todos os participantes do sistema tomam conhecimento sobre a mesma. Uma transação é na verdade uma

mensagem constituída por três partes principais: uma entrada, que é um registro que inclui o *endereço bitcoin* de origem (i.e., o Cliente); um montante, que é a quantidade de *bitcoins* a ser enviada; e uma saída, que é o *endereço bitcoin* de destino, ou seja, o *endereço bitcoin* de quem vai receber o montante a ser enviado (i.e, o Fornecedor).

É preciso dizer que o registro do *endereço bitcoin* de origem não é exclusivamente o *endereço bitcoin* daquele que possui o montante a ser enviado na transação em curso, i.e., do Cliente, mas inclui o *endereço bitcoin* a partir do qual este Cliente recebeu o montante que ele agora deseja enviar para certo participante do sistema (neste caso, o Fornecedor).

Antes de a transação ser submetida, ou seja, transmitida pela Rede, ela é digitalmente assinada usando a chave privada do Cliente. A história da transação passa a ser conhecida por todos participantes da rede, sendo possível de ser completamente averiguada até o ponto de saber onde os *bitcoins*, referentes ao montante enviado na transação, foram originalmente produzidos.

Por fim, ainda a partir da Figura 1, tem-se a atividade *Aguardar Confirmação*. Esta atividade nada mais é do que esperar que a Rede valide a transação executada e informe aos interessados, isto é, ao Cliente e ao Fornecedor. Nas subseções seguintes, esse entendimento é melhor detalhado.

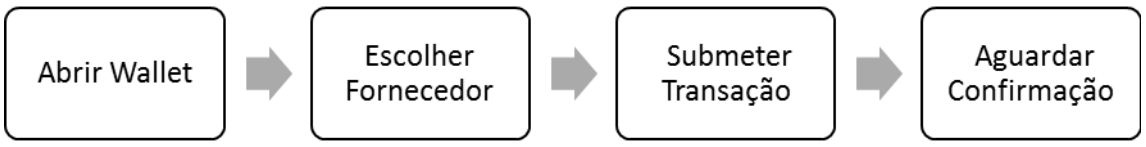

Figura 1. Atividades do Cliente

4.2. Atividades da Rede

A partir da Figura 2, percebem-se as atividades a serem desencadeadas pela Rede, a qual possui uma arquitetura *P2P* (Kurose e Ross 2013). A primeira atividade refere-se a *Verificar Transação*. Essa é a atividade mais importante e complexa da operação do sistema *Bitcoin*.

Toda transação realizada é enviada pela rede por meio de *broadcast* (difusão). Nós específicos da rede, os chamados mineiros, juntam as novas transações que chegam até eles em um mesmo bloco de transações. Esse bloco não permite transações em conflito e cada mineiro trabalha para encontrar a *prova de trabalho* (ou *proof-of-work*) para o mesmo.

Encontrar a *prova de trabalho* para um bloco de transações significa encontrar um *hash* criptográfico atendendo a certo critério. A dificuldade desse critério é continuamente ajustada com base na frequência com que os blocos vão sendo adicionados à *blockchain*. São testadas várias possibilidades até que se tenha a sorte de identificar aquela que funciona. Esse processo de tentativas sucessivas constitui o trabalho da mineração.

Matematicamente, a mineração deve encontrar o valor *nonce* de tal sorte que seja satisfeita a desigualdade expressa por: *SHA-256*(*SHA-256*(*data* + *nonce*)) < *T*, onde [BITCOIN STACKEXCHANGE 2016]:

- *nonce* é um número inteiro que o mineiro escolhe livremente. Determinar, por meio de tentativas sucessivas, o valor de *nonce* capaz de satisfazer a desigualdade estabelecida (< *T)* constitui o trabalho computacional a ser realizado. O valor de *nonce* resultante é denominado de *golden nonce*;
- *data* é o *hash* do conteúdo constituído pelo bloco de transações a ser validado e pelo *hash* do último bloco da *blockchain* atual da rede;
- *SHA-256* é um algoritmo de função *hash* (vide Seção 2.3);
- *T* é um valor ajustado consensualmente pelos nós da rede para garantir que o trabalho computacional seja tal que um bloco de transações é encontrado (validado) a cada 10 (dez) minutos em média.

Um *hardware* moderno pode realizar centenas de milhões de tentativas por segundo (ou, equivalentemente, centenas de milhões de *hashes* por segundo) para encontrar o valor adequado para *nonce* (i.e., *golden nonce*). Para ser competitivo nessa busca, o mineiro necessita, portanto, de *hardware* cada vez mais especializado (i.e., *ASICs - Application Specific Integrated Circuits*), caso contrário, haverá uma tendência de se gastar tanto em consumo de energia elétrica que a recompensa (para o mineiro) não valerá a pena.

Quando o *Bitcoin* foi lançado, a recompensa por bloco minerado era de 50,0 BTC. Esse valor de recompensa é dividido por 2 a cada 230.000 blocos minerados na rede ou, aproximadamente, a cada quatro anos, já que cada bloco leva em média cerca de dez minutos para ser minerado. Esse ajuste é conhecido como *halving*. O *Bitcoin* já passou por um *halving* e, atualmente, cada bloco minerado é recompensado com 25,0 BTC.

Como é impossível encontrar o *golden nonce* sem realizar sucessivas tentativas de valores, envolvendo o cálculo de duas funções *hash*, então se admite que saber o *golden nonce* é a prova necessária e suficiente de que o trabalho computacional obrigatório foi efetivamente realizado ou, em outras palavras, saber o *golden nonce* é a *proof-of-work*.

Toda vez que um bloco de transações é validado, ele deve ser divulgado pela rede e adicionado a uma ramificação da *blockchain* corrente, denominada de *branch*, e os mineiros posteriores vão sempre escolhendo a *branch* de maior comprimento (i.e., a mais longa). A convergência para a *branch* a ser definitivamente aceita (i.e., aquela de maior comprimento entre todas existentes) ocorre em aproximadamente seis blocos ou, em uma hora, tendo em vista que cada bloco leva cerca de dez minutos para ser validado [Nielsen 2013].

Do exposto, note que o problema do gasto em duplicidade é resolvido devido à inclusão do *hash* do último bloco da *blockchain* atual do sistema no parâmetro *data*, o que propicia a interligação do bloco de transação a ser validado com os blocos de

transações já realizadas no sistema, e ao fato de que os mineiros sempre referenciam à *blockchain* (ou *branch*) mais longa existente no sistema.

Neste contexto, um ataque fraudulento significaria calcular e divulgar uma *branch* que necessariamente teria de ser mais longa que a *branch* de fato correta, a qual contém a transação que deveria ser desfeita. Devido ao trabalho matemático envolvido, essa disputa (i.e., uma corrida para mais rapidamente calcular e divulgar a *branch*) poderia ser vencida pelo fraudador apenas se este tivesse uma capacidade computacional maior do que a de todos mineiros *honestos* da rede.

No entanto, usar uma maior capacidade computacional para uma mineração honesta tende a ser muito mais lucrativo, devido às recompensas existentes, do que usar essa capacidade para um ataque fraudulento visando a um pagamento em duplicidade. Daí, o ataque fraudulento tem uma probabilidade desprezível de ocorrência.

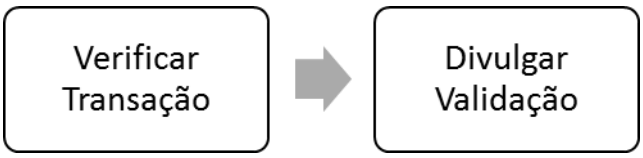

Figura 2. Atividades da Rede

Por convenção, a primeira transação em um bloco é uma transação especial que gera uma nova moeda, de propriedade do criador do bloco. Isso dá um *valor de incentivo* aos nós para manter a rede, e constitui uma forma inicial de colocar novas moedas em circulação, pois não há uma autoridade central para emiti-las.

A adição constante de certo montante de novas moedas no sistema é análoga ao trabalho realizado em *minas de ouro*, onde há mineiros garimpando ouro (dispendendo recursos físicos) para ser posto em circulação. No caso do sistema *Bitcoin*, os recursos dispendidos são tempo de processamento (gasto pelo *hardware*) e eletricidade (i.e., consumo de energia elétrica do *hardware*). O incentivo também pode vir na forma de taxa por transação. Neste caso, o valor de entrada da transação é menor que o valor de saída da mesma, e essa diferença é adicionada ao *valor de incentivo* do bloco que contém essa transação.

Por fim, a partir da Figura 2, tem-se a atividade *Divulgar Validação*. Esta atividade nada mais é do que propagar o bloco validado (incluindo o valor explícito do *golden nonce*), o qual passará a fazer parte da nova *blockchain* do sistema, conforme detalhado anteriormente na explicação da atividade *Verificar Transação*.

4.3. Atividades do Fornecedor

A partir da Figura 3, percebem-se as atividades a serem desencadeadas pelo Fornecedor. A primeira atividade refere-se a *Escutar Transação*. O Fornecedor observa periodicamente sua própria *wallet* e verifica quando seu saldo (balanço financeiro) é alterado por um depósito e a identidade de quem o altera. Isso é feito a partir da inspeção da *blockchain* corrente do sistema e do emprego de assinaturas digitais.

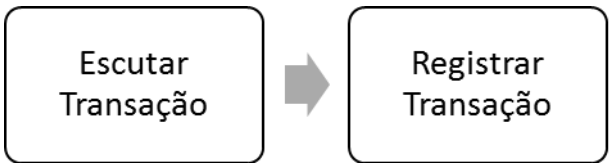

Figura 3. Atividades do Fornecedor

A difusão de blocos recém validados e, consequentemente, das transações nele contidas não necessariamente alcançam todos os nós da rede. Se um nó não receber um bloco, ele requisitará o mesmo quando o próximo bloco for recebido e ele então perceber que o bloco anterior foi perdido.

Ainda, para que um depósito apareça disponível na *wallet* de um participante do sistema (p. ex., o Fornecedor), se faz necessário que o mesmo seja confirmado seis vezes. Isso advém da necessidade de espera para a convergência da *branch* a ser definitivamente aceita para constituir a *blockchain*, o que leva um tempo de aproximadamente 1 hora, conforme já explicado na atividade *Verificar Transação* (vide subseção anterior).

Veja que os *endereços bitcoin* representam a única informação usada para definir onde os *bitcoins* estão e para onde eles são enviados. Esses endereços, como mencionado, são criados de forma privada quando da abertura das *wallets*, garantindo-se anonimato de quem os cria. Entretanto, uma vez que os endereços *bitcoin* são utilizados para realizar transações, eles se tornam vinculados à história de todas as transações (por meio da *blockchain*) em que tenham sido utilizados.

Qualquer participante do sistema pode então ver o saldo e todas as transações de qualquer *endereço bitcoin*. Como o Cliente usualmente tem de revelar sua identidade para poder receber seu produto (ou serviço), seu *endereço bitcoin* não permanecerá completamente anônimo. Para garantir-se o anonimato, se desejado, o Cliente deve então utilizar um *endereço bitcoin* distinto para cada transação que realizar.

Por fim, a partir da Figura 3, tem-se a atividade *Registrar Transação*. Esta atividade não pertence exatamente ao sistema *Bitcoin* em si, tendo sido aqui considerada mais para efeito de completeza da descrição de todo o processo em uma situação real. Nessa atividade admitem-se incluídas, por exemplo, as medidas logísticas para que o produto seja fisicamente entregue ao Cliente.

5. Análise de performance

5.1. Modelagem matemática

Uma transação falsa está restrita à tentativa de o fraudador cancelar um pagamento que ele mesmo realizou e, assim, ter o montante de volta ao seu saldo na *wallet*. Não é possível criar um valor do nada ou que fraudador consiga um montante que nunca pertenceu antes a ele mesmo. Os nós da rede não aceitam transações inválidas como pagamento, e os nós honestos nunca aceitam blocos contendo essas transações [Nakamoto 2008].

Dentro deste contexto, esta seção tem o objetivo de avaliar o nível de segurança provido pelo sistema *Bitcoin*. Para tanto, é considerada a análise de um cenário em que um suposto fraudador busca introduzir uma transação falsa na *blockchain*. Mais especificamente, faz-se a análise de um fraudador tentando gerar uma *branch* mais rapidamente que um mineiro honesto.

A disputa (corrida) entre uma *branch* honesta e uma *branch* fraudulenta pode ser modelada a partir de um Passeio Aleatório Binomial (em inglês, *Binomial Random Walk*). O evento de sucesso é a *branch* honesta ser estendida por 1 bloco, aumentando sua vantagem (em comprimento) em +1, e o evento de fracasso é a *branch* fraudulenta sendo estendida por 1 bloco, reduzindo sua desvantagem (em comprimento) em -1. [Ross 1996; Trivedi 2002].

Decorre então que a probabilidade de uma *branch* fraudulenta compensar uma certa desvantagem inicial com relação a uma *branch* honesta, conseguindo alcançá-la em comprimento após um número infinito de tentativas, é análoga a um problema de Ruína do Jogador (em inglês, *Gambler´s Ruin problem*), que é um clássico exemplo de problema que envolve um Passeio Aleatório (em inglês, *Random Walk*). Essa probabilidade pode ser calculada pela Equação 1 a seguir [Nakamoto 2008].

$$q_z = \begin{cases} 1, & se\ p \leq q \\ \left(\frac{q}{p}\right)^z, & se\ p > q \end{cases} \quad (1)$$

Onde: p é a probabilidade de um mineiro honesto encontrar (resolver) o próximo bloco; q é a probabilidade de o fraudador encontrar (resolver) o próximo bloco; e q_z é a probabilidade de o fraudador eventualmente compensar a desvantagem inicial de estar z blocos atrás.

Agora considere o tempo que o receptor (p. ex., um Fornecedor) de uma nova transação precisa esperar até que se sinta suficientemente seguro de que o emissor (p. ex., um Cliente) não pode mais alterar a transação. Assuma que o emissor é na verdade um fraudador que tenciona enganar o receptor. Neste caso, o emissor submeterá uma transação com o pagamento correto (i. e., uma transação legítima) para o receptor e, após algum tempo, submeterá uma outra transação (i.e., uma transação fraudulenta) usando os mesmos *bitcoins* da transação legítima, realizando um pagamento para ele mesmo (p. ex., direcionando para um outro *endereço bitcoin* pertencente a ele também).

O receptor será eventualmente alertado sobre o outro pagamento do emissor, mas ele (emissor) espera que esse alerta ocorra tarde demais para o receptor, de tal sorte que, por exemplo, o produto relativo à transação já tenha sido enviado para ele (emissor). Como ambas as transações (legítima e fraudulenta) são propagadas por toda a rede, cada nó irá aceitar a primeira transação que veem e rejeitar a segunda. Para que a fraude tenha sucesso, o receptor deve primeiro receber a transação legítima, considerando o pagamento válido. Já a transação fraudulenta deve ser vista pela primeira vez por uma porção significativa do resto dos nós da rede.

Uma vez que todos os nós tenham recebido ou a transação legítima ou a transação fraudulenta, a transação vencedora será determinada pelo mineiro que primeiro realizar a *proof-of-work* do bloco contendo essa transação e conseguir fazer

com que este bloco seja agregado à *blockchain*. Se o fraudador tiver sorte, a transação vencedora irá então conter a transação fraudulenta.

Para fins de evitar uma possível fraude, antes de enviar o produto para o emissor (i.e., Cliente), o receptor (i.e., Fornecedor) deve então esperar até que a transação (pagamento do Cliente) seja adicionada a um bloco e que z blocos posteriores sejam adicionados depois desse bloco.

Obviamente, o receptor não sabe exatamente quanto progresso (ou seja, quantos blocos) um suposto fraudador pode ter conseguido minerar durante essa espera, mas assumindo que blocos corretos (honestos) levaram o tempo médio esperado por bloco individual, o progresso do fraudador pode ser modelado por uma variável aleatória X com distribuição de Poisson com valor médio α calculado pela Equação 2 a seguir [Nakamoto 2008].

$$\alpha = z\frac{q}{p} \tag{2}$$

A Equação 2 informa então quantos blocos o fraudador secretamente conseguiu minerar no mesmo intervalo de tempo em que z blocos honestos posteriores foram calculados. Esse valor é uma função de z e da capacidade relativa do fraudador em relação à capacidade de um mineiro honesto encontrar um bloco.

Por meio das Equações 1 e 2, pode-se então responder a seguinte questão: qual a probabilidade de um fraudador compensar uma diferença estabelecida por ele ter esperado a mineração de z blocos honestos, considerando que esse fraudador já minerou k blocos secretamente durante essa espera?

Para tanto, deve-se realizar o somatório da probabilidade de o fraudador já ter conseguido minerar k blocos até este instante de tempo multiplicada pela probabilidade de ele ainda conseguir minerar os $z - k$ blocos faltantes, para $k = 0, 1, 2, ..., \infty$. Essa modelagem se baseia no Teorema da Probabilidade Total [Trivedi 2006; Ross 1996] e pode ser expressa através da Equação 3 a seguir [Nakamoto 2008].

$$\sum_{k=0}^{\infty} \frac{\alpha^k e^{-\alpha}}{k!} * \begin{cases} q/p^{z-k} & se\ k \leq z \\ 1 & se\ k > z \end{cases} \tag{3}$$

A Equação 3 pode ainda ser reescrita como mostrada a seguir para evitar-se o somatório até o infinito.

$$1 - \sum_{k=0}^{z} \frac{\alpha^k e^{-\alpha}}{k!} \left[1 - \left(\frac{q}{p}\right)^{z-k}\right] \tag{4}$$

5.2 Resultados

Os resultados numéricos calculados por meio da Equação 4 e discutidos a seguir permitem reflexionar sobre a pergunta posta ao final da subseção anterior, a saber: qual a probabilidade de um fraudador compensar uma diferença, medida em número de blocos, estabelecida na *blockchain* devido a ele ter esperado a mineração de z blocos honestos, sendo que esse fraudador já minerou $k \leq z$ blocos secretamente durante essa espera e, ainda, sendo q a probabilidade que ele (fraudador) tem de conseguir minerar o próximo bloco da *blockchain*?

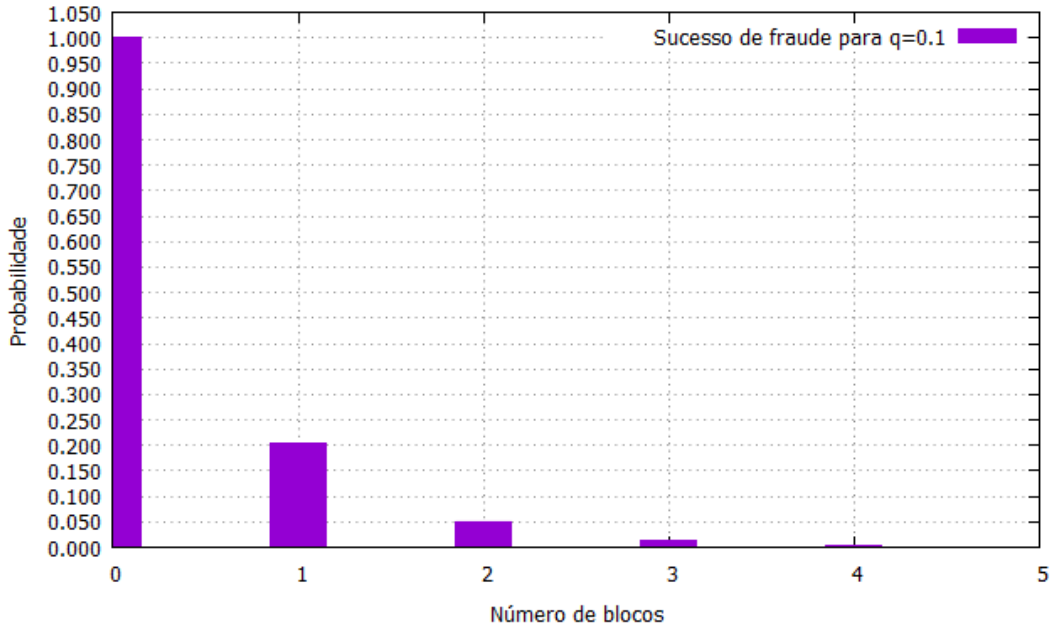

Figura 4. Probabilidade de sucesso de fraude para q = 0,1

Considerando inicialmente uma avaliação ampla do cenário de análise e admitindo q = 0,1, é possível notar que, a partir da Figura 4, a probabilidade de sucesso de fraude decai exponencialmente em função do número de blocos z acrescidos à *blockchain* durante a espera. Mais precisamente, para z = 3, essa probabilidade é de $131{,}7 \times 10^{-4}$, e para z = 6, essa probabilidade é de apenas $24{,}28 \times 10^{-5}$. Isso traz indícios de que a espera média por uma confirmação da transação, admitida como sendo de 6 (seis) blocos minerados ou aproximadamente 1 hora (vide Seção 4.2), tende a ser mais que suficiente.

Considerando-se agora as Figuras 5 e 6 conjuntamente, observa-se o seguinte. Mesmo que o fraudador venha a ter uma probabilidade significativa de sucesso quando da mineração do próximo bloco a ser acrescido à *blockchain*, a espera média usual de 6 (seis) blocos para se ter a confirmação da transação fornece indícios contundentes para admitir-se como suficiente para inibir a fraude.

Mais precisamente, somente ultrapassa-se o limiar de 50% de chance de sucesso quando o fraudador possui uma probabilidade de conseguir minerar o próximo bloco superior a cerca de 0,4 (para z = 3 ou 6); além disso, para z = 6, mesmo para valores significativos de q, a probabilidade de sucesso de fraude ainda é baixa. Por exemplo, para q = 0,25 a probabilidade de fraude é de apenas 0,045. Isso também traz indícios de

que a espera média por uma confirmação da transação, admitida como sendo de 6 blocos minerados ou aproximadamente 1 hora (vide Seção 4.2), tende a ser mais que suficiente.

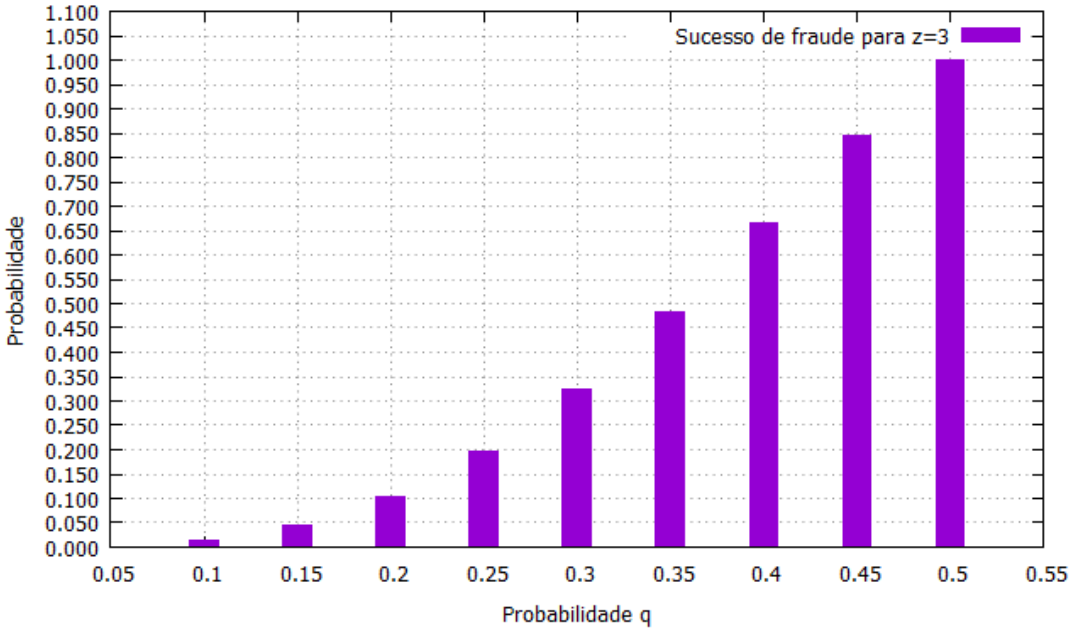

Figura 5. Probabilidade de sucesso de fraude para *z* = 3

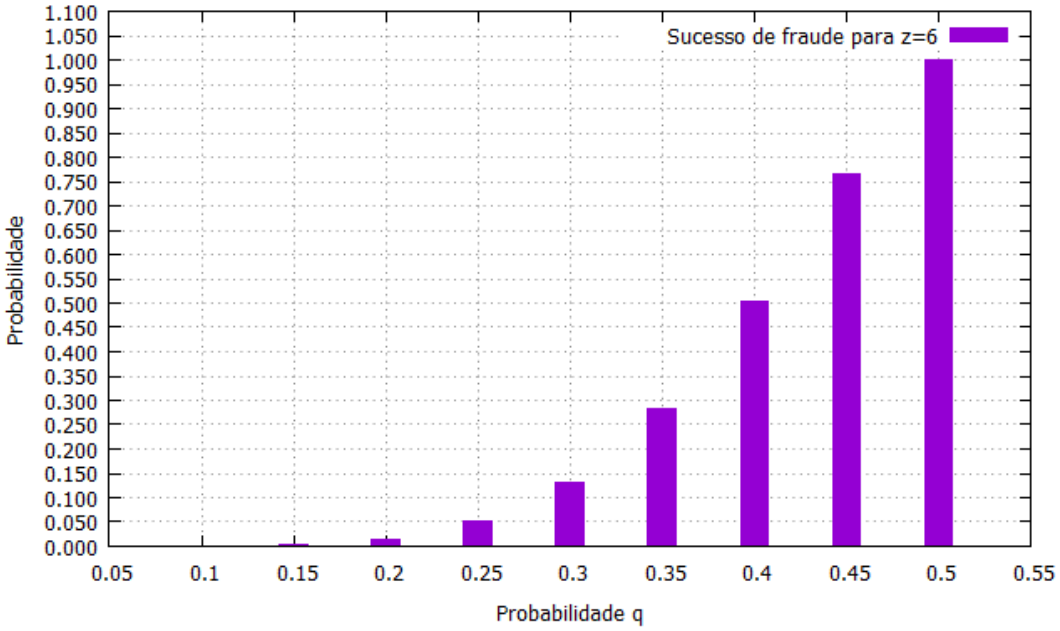

Figura 6. Probabilidade de sucesso de fraude para *z* = 6

5.3. Hipótese, suposições e limitações

Os resutados numéricos alcançados e discutidos nos experimentos demonstram contundentemente a força de segurança das transações do sistema *Bitcoin*. No entanto, por dever de justiça e dando-se benefício à dúvida, é preciso destacar o que se segue.

Em que pese a perfeita razoabilidade de se supor, como cenário de investigação suficiente, a hipótese da disputa para adição de blocos minerados na *blockchain*, envolvendo um nó honesto e um nó fraudador, para mensuração do nível de segurança do sistema, é possível conjecturar que a modelagem analítica apresentada pode não ser suficientemente perfeita em todas as peculiariedades comportamentais do sistema, especialmente quanto à diversidade e simultaneidade de eventos.

Ante o exposto, o desenvolvimento de modelos de simulação bem como a realização de experimentos de medição em sistemas reais, quando submetidos a condições de *stress* de eventos (transações), seriam ambos recomendáveis para fins de complementação e ratificação dos resultados aqui alcançados.

6. Conclusões Finais e Trabalhos Futuros

Este artigo teve o objetivo principal de analisar o nível de segurança associado às transações do sistema *Bitcoin*. Para tanto, as atividades que integram a execução de uma transação no sistema foram cuidadosamente analisadas. Em seguida, através de modelagem matemática, foram realizados experimentos para avaliar o nível de segurança de uma transação ordinária do sistema.

Dentre os resultados numéricos alcançados, pôde-se principalmente confirmar que a espera média por uma confirmação da transação, admitida como sendo de seis blocos minerados, ou aproximadamente 1 hora, é mais que suficiente para garantir-se a segurança do sistema. Além disso, por dever de justiça e dando-se benefício à dúvida, é preciso destacar que a modelagem analítica aqui apresentada pode não ser suficientemente perfeita em todas as peculiariedades comportamentais do sistema, especialmente quanto à diversidade e simultaneidade de eventos, refletindo a inexorável dinâmica dos sistemas reais. Neste contexto, a utilização de modelos de simulação bem como a realização de experimentos de medição em sistemas reais poderiam ser ambos utilizados para fins de se complementação dos resultados aqui alcançados.

Como trabalhos futuros, admite-se inicialmente a reflexão a seguir. A definição da *blockchain* não se importa se 1,0 BTC representa certo montante de dólares ou de qualquer outra moeda. Os clientes envolvidos na transação podem decidir o que 1,0 BTC efetivamente representa. O sistema *Bitcoin* envolve, portanto, muito mais que simplesmente dinheiro e pagamentos, passando a incluir uma infinidade de propriedades associadas a transações. Esse sistema permite inclusive a implementação de *contratos inteligentes* [Chávez e Rodrigues 2016], e, assim, a automatização do fluxo de transferências de dinheiro [WEUSECOINS 2016].

Ethereum [Wood 2015; ETHEREUM HOMESTEAD 2016] constitui um ambiente real para exemplificação da reflexão dada no parágrafo anterior. É um ambiente que flexibiliza a definição semântica do conteúdo a ser guardado na base de dados, i.e., na *blockchain*. Isso com o intuito de prover uma plataforma geral de *contratos inteligentes* sem a dependência de terceiras partes confiáveis para validação e

controle das transações realizadas. Comparativamente aos tradicionais modelos de gestão orçamentária e financeira, passa a existir uma expectativa de redução da signifitiva burocracia e complexidade hoje existentes no Brasil [Feijó et al. 2014; Feijó e Ribeiro 2014; BRASIL 2015].

Por fim, ante a reflexão anterior, recomendam-se então os seguintes dois caminhos como trabalhos futuros. Primeiro, o estudo e a análise do emprego do ambiente *Ethereum* para automatização da gestão orçamentária e financeira de um pequeno município do Brasil, mas mantendo-se em simultâneo o modelo já vigente. Isso permitiria avaliar a expectativa de redução da burocracia existente, bem como da segurança do novo modelo automatizado. Segundo, o estudo e a análise do emprego do ambiente *Ethereum* para a gestão orçamentária e financeira do setor privado do Brasil, mas também se mantendo em simultâneo o modelo vigente de gestão. Como antes, isso permitiria, por simples comparação, ratificar a expectativa de redução da burocracia existente, bem como da segurança do novo modelo automatizado.

Referências

BITCOIN. Disponível em: https://bitcoin.org/en/faq#how-does-bitcoin-work. Acesso em: jul. 2016.

BITCOIN SIMPLIFIED. Disponível em: http://bitcoinsimplified.org/. Acesso em: jul. 2016.

BITCOIN STACKEXCHANGE. What exactly is mining. Disponível em: http://bitcoin.stackexchange.com/questions/148/what-exactly-is-mining. Acesso em: jul. 2016.

BRASIL (2015). Ministério do Planejamento, Orçamento e Gestão. Secretaria de Orçamento Federal. Manual técnico de orçamento MTO. Edição 2016. Brasília.

Chávez, J. J. G.; Rodrigues, C. K. S. (2015). Hopping among Pools in the Bitcoin Mining Network. The SIJ Transactions on Computer Networks & Communication Engineering (CNCE), v. 3, n. 2, p. 22-27.

Chávez, J. J. G.; Rodrigues, C. K. S. (2016). Automatic hopping among pools and Distributed application. In: XXI Sysmposium on Signal Processing, Images and Artificial Vision (STSIVA). Bucaramanga, Colombia, August.

COINDESK. Disponível em: http://www.coindesk.com/price/. Acesso em: jul. 2016.

Dyhrberg, A. H. (2016). Bitcoin, gold and the dollar – a GARCH volatility analysis. Finance Research Letters, v. 16, p. 85-92, February.

Dwyer, G. P. (2011). The economics of Bitcoin and similar private digital currencies. Journal of Financial Stability, v. 8, p. 81-91, April.

ETHEREUM HOMESTEAD. Ethereum Homestead Documentation. Disponível em: http://www.ethdocs.org/en/latest/. Acesso em: jul. 2016.

Feijó, P. H.; Pinto, L. F.; Mota, F. G.; Da Silva, L. C. (2014). Curso de SIAFI: Uma abordagem prática da execução orçamentária financeira. 3a. ed., v. I, Brasília: Gestão Pública.

Feijó, P. H.; Ribeiro, C. E. (2014). Entendendo o Plano de Contas Aplicado ao Setor Público: PCASP – Exercícios e Estudo de Caso com Lançamentos Típicos. Série Entendendo CASP. 1ª ed., Brasília: Gestão Pública.

Feld, S.; Schönfeld, M.; Martin, W. (2014). Analyzing the deployment of Bitcoin's P2P network under an AS-level perspective. Procedia Computer Science, v. 32, p.1121–1126.

Kiayias, A.; Panagiotakos, G. On trees, chains and fast transactions in the blockchain, 2016. Disponível em: http://eprint.iacr.org/2016/545. Acesso em: jul. 2016.

Kurose, J. F.; Ross, K. W. (2013). Computer networking: A top-down approach featuring the Internet. Pearson Education, 6th ed., New York.

Luther, W. J. (2015). Bitcoin and the Future of Digital Payments. Disponível em: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2631314. Acesso em: fev. 2016.

Nakamoto, S. (2008). Bitcoin: A Peer–to–peer Electronic Cash System. Disponível em: http://www.bitcoin.org/bitcoin.pdf. Acesso em: jul. 2015.

Nielsen, M. (2013). How the Bitcoin protocol actually works. Disponível em: http://www.michaelnielsen.org/ddi/how-the-bitcoin-protocol-actually-works/. Acesso em: fev. 2016.

Pazmiño, J. E.; Rodrigues, C. K. S. (2015). Simply Dividing a Bitcoin Network Node may Reduce Transaction Verification Time. The SIJ Transactions on Computer Networks & Communication Engineering (CNCE), v. 3, n. 2, p. 17-21.

Rocha, J. G.; Rodrigues, C. K. S. (2016). O Processo de Negócio do Sistema de Transações Financeiras Bitcoin. Universitas: Gestão e TI, v. 6, n. 1, p. 1-10, julho.

Rosenfeld, M. (2011). Analysis of Bitcoin Pooled Mining Reward Systems. Disponível em: http://arxiv.org/pdf/1112.4980v1.pdf. Acesso em: jul. 2016.

Ross, S. M. (1996). Stochastic Processes. John Wiley & Sons, Inc., 2nd ed., New York.

Roth, N. (2015). An architectural assessment of Bitcoin: using the System Modeling Language. Procedia Computer Science, v. 44, p. 527-536.

Silva, G. A. B.; Rodrigues, C. K. S. (2016). Mineração individual de bitcoins e litecoins no mundo. In: XVI Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais. Workshop de Trabalhos de Iniciação Científica e de Graduaçao (SBSeg), Niterói, RJ, novembro, p. 524-533.

Tanenbaum, A. S.; Wetherall, D. J. (2010). Computer networks. Pearson Education, 5th ed., New York.

Trivedi, K. S. (2002). Probability and Statistics with Reliability, Queuing and Computer Science Applications. John Wiley & Sons, Inc., 2nd ed., New York.

WEUSECOINS. Disponível em: https://www.weusecoins.com/. Acesso em: jul. 2016.

Wood, G. (2015). Ethereum: A secure decentralised generalised transaction ledger. Disponível em: http://tech.lab.carl.pro/kb/ethereum/yellowpaper. Acesso em: jul. 2016.