Brazilian Journal of Development

Analisando a segurança das transações do sistema de pagamento eletrônico IOTA

Analyzing transaction security of the IOTA electronic payment system

DOI:10.34117/bjdv5n9-057

Recebimento dos originais: 18/07/2019
Aceitação para publicação: 10/09/2019

Guilherme Gouveia Martinez
Graduando em Ciência da Computação pela Universidade Federal do ABC.
Instituição: Centro de Matemática, Computação e Cognição (CMCC) da Universidade Federal do ABC (UFABC).
Endereço: Avenida dos Estados, 5001, Bloco A, Santa Terezinha, Santo André – SP, Brasil.
E-mail: martinez.guilherme@outlook.com.br

Carlo Kleber da Silva Rodrigues
Doutor em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro.
Instituição: Centro de Matemática, Computação e Cognição (CMCC) da Universidade Federal do ABC (UFABC).
Endereço: Avenida dos Estados, 5001, Bloco A, Santa Terezinha, Santo André – SP, Brasil.
E-mail: carlo.kleber@ufabc.edu.br

RESUMO

Este trabalho analisa a segurança das transações realizadas no sistema de pagamento eletrônico IOTA. Para tanto, um estudo das etapas constitutivas do processo de execução de transações é primeiramente conduzido. Na sequência, por meio de modelagem matemática e simulação, são realizados experimentos para estimar a probabilidade de fraude, caracterizada pela alteração da informação já registrada no sistema. Os resultados finais demonstram a atrativa robustez do sistema. Por exemplo, mesmo que um atacante consiga controlar 40% do total de transações entrantes no sistema, a probabilidade de fraude é estimada em apenas 0,03. Por fim, conclusões gerais e trabalhos futuros concluem este artigo.

Palavras-chave. IOTA, Tangle, Segurança, Tecnologia de Registros Distribuídos.

ABSTRACT

This paper analyzes the security of the transactions performed in the IOTA electronic payment system. To this end, a study of the constitutive steps of the transaction execution process is first conducted. Subsequently, through mathematical modeling and simulation, experiments are performed to estimate the probability of fraud, characterized by the alteration of the information already recorded in the system. The final results demonstrate the attractive robustness of the system. For example, even if an attacker can control 40% of the total incoming transactions in the system, the probability of fraud is estimated at only 0.03. Finally, general conclusions and future works conclude this article.

Keywords. IOTA, Tangle, Security, Distributed Ledger Technology.

1. INTRODUÇÃO

O sistema IOTA [Popov 2018a] é um sistema de pagamento eletrônico baseado em criptomoedas. A denominação da criptomoeda utilizada é a mesma dada ao sistema, diferenciando-se apenas por ser escrita com letras minúsculas, i.e., criptomoeda *iota*. Esta criptomoeda foi originalmente concebida para a indústria da Internet das Coisas (do inglês, *Internet of Things* – IoT), visando fulcralmente atender a aplicações executadas por dispositivos computacionais de baixo poder de processamento e contemplando transações de valores financeiros reduzidos [IOTA 2019; Choudhary et al. 2020].

As criptomoedas constituem uma alternativa digital para as moedas fiduciárias já existentes no mundo. Para fins de garantir a segurança das transações, bem como o anonimato das partes envolvidas, os sistemas de pagamento baseados em criptomoedas utilizam métodos criptográficos. Além disso, esses sistemas têm a sua administração feita sob um paradigma inteiramente descentralizado, permitindo prescindir da existência de órgãos formais centralizados (e.g., bancos públicos e privados) para fins de regulação ou validação das transações realizadas [Sallal 2018; Luther 2015].

A ausência de regulação por parte de órgãos centralizados faz com que o valor das criptomoedas obedeça à lei da oferta e da procura [Iwamura et al. 2014], resultando em flutuações ao longo do tempo. Por exemplo, a criptomoeda *iota* foi criada em 2016, tendo sido disponibilizada na plataforma de negociação de criptomoedas *Bitfinex* em 2017, com um valor inicial de USD 0,25. No período de novembro até dezembro de 2017, seu valor sofreu um aumento de aproximadamente 1.800%, quando chegou a valer USD 5,80, mas em seguida perdeu cerca de ¼ de seu valor. Atualmente, um *iota* está cotado em cerca de USD 0,31 [CoinMarketCap 2019].

Além do sistema IOTA, alvo de pesquisa deste trabalho, existem outros sistemas de criptomoedas como, por exemplo, Ethereum, XRP, Litecoin, Zcash e Bitcoin [Nakamoto 2008; CoinMarketCap 2019; Yeow et al. 2017; Wood 2015]. Cada um deles possui características diferentes que os tornam mais adequados para certos cenários de aplicação. Mas, particularmente para aplicações IoT, o sistema IOTA tem sido majoritariamente assinalado como o projeto mais atrativo, pois não exige um alto poder de processamento por parte dos dispositivos computacionais e tampouco demanda taxas para validação de transações. Note que essas duas características democratizam o uso do sistema, pois qualquer dispositivo computacional pode ser parte integrante do sistema e, mais importante, qualquer

transação, independentemente de seu valor financeiro, é economicamente viável de ser processada [Mata e Rodrigues 2019].

Dentro deste contexto, este artigo tem como principal objetivo analisar a segurança das transações realizadas no sistema IOTA. Para tanto, primeiramente é apresentada uma explicação geral do sistema, considerando cada atividade que integra a execução e a validação de uma transação. Em seguida, por meio de modelagem matemática e simulação, são feitos experimentos para avaliar a segurança das transações executadas no sistema, considerando a ocorrência de fraudes originadas pela alteração de informações já registradas na sua base de dados.

Ante o objetivo a ser alcançado, este artigo tem, como principal contribuição, a disponibilização de um único texto que provê um entendimento das atividades constituintes da execução de uma transação no sistema IOTA, bem como uma análise formal da segurança das transações realizadas. Salvo melhor juízo, este é o primeiro trabalho da literatura científica a promover essa discussão fazendo uso conjunto de modelagem matemática e simulação.

A organização do restante deste artigo é descrita a seguir. A Seção 2 explica o que vem a ser uma tecnologia de registros distribuídos. Essa explicação traz uma base conceitual para o adequado entendimento do sistema IOTA, tornando este artigo o mais autocontido possível. Os trabalhos relacionados são discorridos na Seção 3. A Seção 4 explica o sistema IOTA. A Seção 5 traz a análise da segurança do sistema IOTA, examinando e discutindo sobre os resultados obtidos nos experimentos. Por fim, conclusões gerais e trabalhos futuros constituem a Seção 6.

2. TECNOLOGIA DE REGISTROS DISTRIBUÍDOS

Uma tecnologia de registros distribuídos (do inglês, *Distributed Ledger Technology* – DLT) consiste na tecnologia utilizada para a implementação das bases de dados dos sistemas de criptomoedas [Ehmke et al. 2019; Kannengießer et al. 2019; Rodrigues e Da Silva 2019; Siris et al. 2019]. Considerando sua definição, os dados desses sistemas são replicados, parcial ou integralmente, em nós processadores geograficamente distribuídos e interligados por meio de uma rede de computadores de arquitetura *peer-to-peer* (P2P). Os dados se referem às informações resultantes das transações realizadas nesses sistemas.

Cada nó processador da rede é capaz de manipular individualmente alterações na base de dados e alcançar as suas próprias conclusões. Uma alteração passa a ser válida se, e somente se, há um consenso entre a maioria dos nós processadores. Estes nós são então atualizados de

maneira independente com as novas informações [Ehmke et al. 2019; Kannengießer et al. 2019; Rodrigues e Da Silva 2019; Siris et al. 2019].

Os registros contidos nessa base de dados distribuída podem ser *fechados*, acessíveis apenas para uma determinada categoria de usuários (i.e., registros que requerem autorização), ou *abertos*, acessíveis por todos os usuários do sistema (i.e., registros que não requerem autorização). Esses registros podem ser utilizados em inúmeros tipos de aplicações computacionais, servindo aos mais diversos propósitos e classes de ativos. De maneira geral, a segurança e a consistência desses registros são obtidas pelo uso de funções de *hash* criptográficas e assinaturas digitais [Ehmke et al. 2019; Kannengießer et al. 2019; Mata e Rodrigues 2019; Siris et al. 2019].

Dentre as DLTs com maior destaque, podem ser citadas as tecnologias *Blockchain* e *Tangle*. *Blockchain* foi originalmente apresentada em conjunto com a proposta do sistema de pagamento eletrônico *Bitcoin.* A base de dados dessa DLT consiste em *blocos* de dados interligados sob uma disposição de *lista encadeada*. Mais precisamente, cada bloco está ligado de forma exclusiva a um único bloco anterior na lista, denominado de bloco *pai*. Cada bloco armazena os dados de $n \approx 500$ transações realizadas no sistema. A adição dos blocos à lista ocorre de forma cronológica [Antonopoulos 2017; Nakamoto 2008; Sallal 2018; Mata e Rodrigues 2019].

Tangle é a DLT utilizada para a implementação da base de dados do sistema de pagamento IOTA [IOTA 2019; Gal 2018]. A base de dados dessa DLT é organizada sob a forma de um grafo acíclico orientado (do inglês, *Direct Acyclic Graph* – DAG). Cada vértice (do inglês, *site*) do DAG armazena os dados de exclusivamente uma única transação realizada no sistema (i.e., $n = 1$). Cada vértice liga-se a dois outros vértices anteriores do DAG [Popov 2018a; Yeow et al. 2017].

3. TRABALHOS RELACIONADOS

Esta seção tem o objetivo de discorrer sobre trabalhos da literatura que se relacionam ou contribuem, direta ou indiretamente, para o objetivo deste trabalho, bem como prover uma visão do estado da arte desta área de pesquisa.

Por meio de modelagem discreta, Kúsmierz (2017) examina as propriedades estruturais de um DAG para implementação da *Tangle*. Em específico, são examinadas as propriedades básicas do *peso cumulativo*, que é um valor proporcional ao número de validações que uma transação recebe, e a estabilidade do número de *pontas* (do inglês, *tips*),

que é o nome dado aos vértices do DAG cujas transações ainda não foram validadas. Os resultados revelam duas fases distintas de crescimento do *peso cumulativo*: exponencial e linear. A primeira refere-se a um período inicial do sistema, conhecido como fase de adaptação, enquanto a última ocorre após o período mencionado. Tais informações são imperiosas para avaliar a segurança do sistema nas duas diferentes fases, pois o *peso cumulativo* é um dos responsáveis por determinar se uma transação terá maior ou menor probabilidade de ser validada. Portanto, ele é fundamental para estimar a probabilidade de uma fraude ocorrer.

Yeow et al. (2018) discutem comparativamente sistemas de consenso descentralizados para aplicações IoT, inspecionando os prós e os contras de cada sistema. São avaliadas as estruturas de dados, a escalabilidade e os modelos de transação. Dentre as tecnologias adotadas para a implementação das estruturas de dados, estão *Blockchain* e *Tangle*. Sobre a primeira, verificam-se principalmente problemas de escalabilidade, devido à baixa taxa de geração de blocos, bem como retardo na propagação de blocos de grande tamanho entre os nós processadores. Já para o segundo, pode haver a necessidade de um controlador centralizado para que seja evitada a denominada tragédia dos comuns (do inglês, *tragedy of the commons*), isto é, evitar que os indivíduos agindo de maneira egoísta prejudiquem o sistema. Além disso, existe a probabilidade de que algumas transações nunca venham a ser aprovadas, tornando-se *pontas* permanentes. Em síntese, esse trabalho conclui que o sistema baseado em *Tangle* é o mais adequado para a IoT.

Bramas (2018) faz uma análise do sistema IOTA, buscando identificar o tempo médio de confirmação para uma transação e estimar o número médio de transações que não foram confirmadas ao longo do tempo, i.e., das *pontas*. Os resultados ressaltam primordialmente a necessidade de um significativo número de nós processadores honestos constantemente validando as transações. Caso contrário, não seria possível proteger a rede de ataques, pois o menor número de nós honestos implicaria transações honestas com menor peso cumulativo quando comparadas com o peso cumulativo de uma transação fraudulenta validada por nós processadores desonestos.

Kusmierz e Gal (2018) analisam a probabilidade de uma dada transação em uma *Tangle* vir a tornar-se uma transação não validada permanente, i.e., uma *ponta permanente*. Para que a probabilidade de uma *ponta* existir seja mantida constante, é preciso reduzir a tendência de o sistema selecionar e validar certas transações. Essa tendência é controlada por um parâmetro de configuração sistêmica denominado de *alfa*. Para valores altos de *alfa*, há

uma maior preferência de o sistema escolher transações antigas (ou já validadas) em detrimento das mais recentes. Por outro lado, a redução desse valor gera um impacto negativo na segurança do sistema, pois há uma randomização na seleção de transações, o que pode facilitar a ação de atacantes. Isso decorreria da menor influência que o peso cumulativo teria na escolha, ocasionando uma maior probabilidade de selecionar-se uma transação fraudulenta ou não segura, i.e., uma transação que recebeu poucas ou nenhuma validação. Sendo assim, esse trabalho destaca a importância de se escolher cuidadosamente o parâmetro *alfa* para que sejam minimizados os efeitos negativos na segurança e, ao mesmo tempo, garantir a operação eficiente do sistema.

Mata e Rodrigues (2019) realizam uma análise competitiva entre as tecnologias *Blockchain* e *Tangle*, com foco em aplicações IoT. A partir de simulações são mensurados os níveis de segurança de ambas tecnologias em diferentes cenários, considerando a ocorrência de fraudes caracterizadas pela substituição de valores válidos de transações anteriores. Os experimentos realizados demonstram uma maior atratividade da tecnologia *Tangle* em relação à *Blockchain*, particularmente pelo fato de a primeira necessitar de ajustes de configuração mais simples para obter maiores níveis de segurança.

A tecnologia *Tangle* foi detalhadamente apresentada no trabalho de Popov (2018a). O autor retrata a filosofia e o mecanismo geral de sua operação. Em especial, também são investigadas as principais maneiras de ataque que a estrutura pode vir a sofrer. Esse trabalho constitui uma das mais importantes referências introdutórias da literatura sobre esse tema. Salienta-se que a primeira versão deste trabalho foi escrita no ano de 2017, e este vem sendo sistematicamente revisado, constituindo-se assim em um valioso *white paper*.

Com a finalidade de evitar prováveis ataques à *Tangle*, Popov (2018b) introduz a ideia de modificações *locais*, as quais consistem em modificações feitas pelos nós processadores nos registros das transações, considerando informações localmente disponíveis. Por exemplo, um nó poderia reduzir o *peso* de uma transação caso ele notasse uma grande diferença na informação de tempo local entre ela e o tempo corrente na rede. Tal ação seria tomada como medida preventiva, pois poderia implicar que a transação foi gerada bem antes por um atacante e estava oculta até aquele momento para, posteriormente, ser usada em uma tentativa de ataque. Tal redução diminuiria a probabilidade de que ela fosse válida e, consequentemente, também as chances de que o responsável pelo ataque obtivesse sucesso. Neste contexto, esse trabalho apresenta então uma maneira eficiente de como, através da análise comportamental da rede, seria possível aumentar a segurança sistêmica.

Cullen et al. (2019) analisam um cenário de ataque em que um *ramo parasita* (do inglês, *parasite branch*) é adicionada à *Tangle*. Um *ramo parasita* consiste em um DAG fraudado, denominado de *Sub-Tangle*, que possui uma transação conflituosa e que eventualmente é adicionado à *Tangle* para fins de fraude. Por exemplo, esse *ramo parasita* pode ter o objetivo de transferir uma certa quantia (montante em criptomoedas) de volta para o atacante, tornando sem efeito outra transação, já presente na *Tangle*, que destinava esse mesmo montante para o vendedor de um produto. Como principal contribuição, os autores propõem novas equações que definem a probabilidade de que uma transação venha a ser selecionada, buscando aprimorar a resistência do sistema a esse tipo de ataque durante a fase de adaptação.

Ante o exposto, em que pesem a diversidade e a importância dos resultados já alcançados na literatura, a contribuição e a diferenciação deste artigo se revelam pelo relativo ineditismo da realização de uma avaliação da segurança quanto à inviolabilidade das informações já registradas no sistema IOTA, sob o viés combinado da modelagem matemática e da simulação.

4. SISTEMA IOTA

Para explicar a operação do sistema IOTA, tem-se a seguir a descrição das atividades constituintes de um processo genérico de aquisição de um produto ou serviço, realizada por um Cliente a um Fornecedor. Para fins de facilidade de exposição, consideram-se separadamente as fases de: solicitação do Cliente; verificação da transação pela Rede (i.e., pelos nós processadores); e confirmação da transação pelo Servidor.

4.1. SOLICITAÇÃO DO CLIENTE

A Figura 1 mostra as atividades a serem desencadeadas pelo Cliente. A primeira atividade refere-se a *Abrir Carteira*. A *carteira* é equivalente a uma conta bancária do sistema financeiro tradicional. Ela permite que o Cliente possa receber, guardar e enviar *iotas* para os outros participantes do sistema [IOTA 2019]. Para *Abrir Carteira*, o Cliente deve inicialmente gerar uma *chave de acesso* (do inglês, *seed*), que é utilizada para acessar a *carteira*. A *chave de acesso* consiste em uma sequência com 81 caracteres de comprimento, devendo conter apenas letras maiúsculas de A a Z e, ainda, o número 9.

Há dois principais tipos de *carteiras: completa* e *parcial* (do inglês, *full node* e *light node*, respectivamente) [IOTA 2019]. No primeiro tipo, o Cliente armazena toda a *Tangle* em seu dispositivo e, também, realiza o trabalho de validação de transações existentes no sistema. O trabalho correspondente à validação de transações é descrito em detalhes na Subseção 4.2. Dessa forma, o Cliente ajuda a aumentar a eficiência do sistema, executando também o papel de um nó processador. No segundo tipo, o Cliente acessa a *Tangle* que está armazenada remotamente em um nó processador, mas não realiza o trabalho de validação de transações no sistema.

Para cada transação realizada (e.g., transferência de um montante em *iotas* para um Fornecedor), o Cliente utiliza três endereços *iota* distintos: um primeiro endereço serve para identificar a origem (i.e., a carteira do Cliente) do montante em *iotas* a ser transferido; um segundo endereço serve para identificar o destino do montante a ser transferido (i.e., a carteira do Fornecedor); um terceiro e último endereço serve para indicar o destino do saldo remanescente da transação (i.e., um novo endereço associado a Carteira do Cliente). Por exemplo, caso o Cliente tenha dez *iotas* armazenados no *endereço A* e queira transferir sete *iotas* para um endereço *B*, um novo *endereço C*, pertencente ao Cliente, seria gerado, e este endereço *C* receberia os três *iotas* não utilizados. Ou seja, o Cliente pode possuir múltiplos endereços *iotas* associados a uma mesma carteira.

Seguindo a Figura 1, a segunda atividade é *Escolher Fornecedor*. Para tanto, o Cliente deve apenas indicar o *endereço iota* associado ao Fornecedor. A terceira atividade consiste em *Submeter Transação*. Ao realizá-la, o Cliente executa a *prova de trabalho* de duas outras transações ainda não validadas. A *prova de trabalho* consiste na resolução de um problema matemático. Este problema requer um significativo processamento computacional, mas é simples de ter sua resolução verificada. Seu objetivo é proteger o sistema contra *spam*s. [Popov 2018a; IOTA 2019].

Mais especificamente, para realizar a supracitada *prova de trabalho*, é necessário encontrar uma variável, denominada de *nonce*, que concatenada com os dados da transação a ser validada, é a entrada de uma função *hash* criptográfica. A saída dessa função deve possuir uma forma particular [Popov 2018a; IOTA 2019]. Em especial, tal saída deve possuir um certo número de zeros. Caso essa condição não seja satisfeita, o valor de *nonce* é incrementado e um nova tentativa é feita. Isso se repete até que a condição seja satisfeita [IOTA 2019].

Cada transação, antes de ser submetida, é assinada digitalmente utilizando criptografia assimétrica. Mais especificamente, cada transação é assinada com uma distinta *chave privada*,

Brazilian Journal of Development

gerada a partir da combinação da *chave de acesso* com um inteiro positivo qualquer (incluindo o zero). Cada chave privada é também utilizada para gerar o endereço *iota* associado à transação envolvida (i.e., por ela assinada), o que resulta em múltiplas tuplas <*chave privada*, *endereço iota*> [IOTA 2019; Welz 2019]. A autoria das transações é, por sua vez, verificável por qualquer outro participante da rede utilizando as correspondentes chaves públicas e assinaturas [Welz 2019]. Menciona-se que esse processo é baseado no *esquema de assinatura única de Winternitz* (do inglês, *Winternitz one-time signature scheme* (W-OTS)) [Buchmann et al. 2011; Welz 2019].

Por fim, ainda a partir da Figura 1, tem-se a atividade *Aguardar Validação*. Esta consiste em esperar que os nós processadores do sistema atinjam um consenso. Nas subseções seguintes, esse entendimento é mais detalhado.

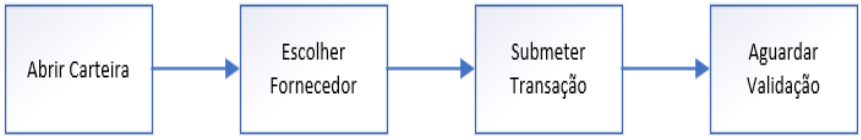
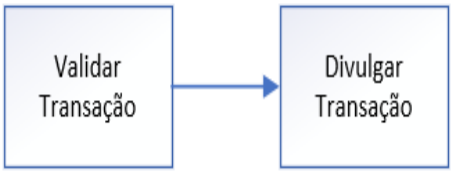


Figura 1. Atividades do Cliente.

4.2. VERIFICAÇÃO DA TRANSAÇÃO PELA REDE

As atividades da Rede consideradas nesta subseção buscam ilustrar como o consenso é atingindo entre os nós processadores e como se dá o trabalho de validação de uma transação. Para tanto, com fins de maior facilidade de exposição, é estabelecida uma divisão artificial entre quem submete transações e quem as valida.

A partir da Figura 2, percebem-se as atividades a serem desencadeadas pela Rede, i.e., pelos nós processadores, a qual possui uma arquitetura *peer-to-peer (*P2P). A primeira atividade é *Validar Transação*. Esta é a mais importante e complexa atividade da operação do sistema IOTA.



Brazilian Journal of Development

Figura 2. Atividades da Rede.

Na Figura 3, para fins de exemplificação, é ilustrado um DAG representativo de uma *Tangle*. Este DAG possui sete vértices e, portanto, sete transações. Lembramos que cada vértice armazena uma única transação. Por facilidade de discussão e referenciação, os termos vértice e transação são doravante utilizados de forma intercambiável, deixando-se subentendida a diferenciação conceitual entre os mesmos.

As sete transações são enumeradas de 0 a 6. Uma transação ao ser inicialmente anexada deve referenciar (graficamente por setas) duas outras transações não validadas (i.e., *pontas*). Essas duas *pontas* são então submetidas à atividade *Validar Transação*. Durante essa atividade, é examinado se o valor a ser transferido pela transação não excede o suprimento global total (i.e., se não são gastas mais criptomoedas do que existem no sistema), se todas as assinaturas são válidas, se a *prova de trabalho* foi realizada, e se a transação não resulta em um balanço negativo a algum endereço. A transação submetida pelo Cliente então se torna uma *ponta*, representada pela cor amarela na Figura 3.

É preciso certo tempo para executar a *prova de trabalho* e difundi-la por toda a Rede. Dessa forma, note que as transações selecionadas para serem validadas podem não ser mais *pontas* quando a sua eventual transação validadora correspondente (i.e., a transação do Cliente) for adicionada à *Tangle*. Consequentemente, uma transação pode ser referenciada por mais de uma transação (e.g., a *transação* 1 na Figura 3).

A partir da Figura 3, vemos também que a transação 5 referencia diretamente as transações 2 e 3, respectivamente, e indiretamente as transações 1 e 0, respectivamente. A transação *gênesis*, representada pela *transação* 0, é referenciada tanto de forma direta como indireta por todas as demais transações da rede. Explica-se que, ao início da operação do sistema IOTA, a *transação* gênesis foi responsável por conter todo o montante de *iotas* do sistema, e então distribuí-lo entre as diversos outras transações [Popov 2018a; Gal 2018]. Por definição de projeto, nenhuma criptomoeda *iota* é possível de ser criada após este início.

Cada transação tem associado um número inteiro proporcional à quantidade de trabalho investido pelo Cliente ao submetê-la, denominado *peso*. A ideia é que as transações com maiores pesos são mais importantes. Outro conceito relevante é o de *peso cumulativo*. O *peso cumulativo* nada mais é que a soma dos pesos de todas as transações que referenciam de maneira direta ou indireta uma transação, somado ao seu próprio peso.

Brazilian Journal of Development

Para simplificar, vamos assumir que todas as transações da Figura 3 possuem peso igual a 1. Logo, o *peso cumulativo* da transação 1 seria igual à soma dos pesos das transações 2, 3, 4, 5 e 6, mais o próprio peso de 1. Assim, temos que o *peso cumulativo* da transação 1 é igual a 6. Seguindo a mesma ideia, a transação 5 teria peso igual a 2, pois ela é referenciada apenas pela transação 6.

O *peso cumulativo*, após um período de adaptação (em que ocorre um crescimento exponencial), passa a crescer linearmente [Kúsmierz 2017] de acordo com a taxa de transações novas (representada por λ) que chegam ao sistema. Ou seja, a cada unidade de tempo, o *peso cumulativo* aumenta, em média, + λ, considerando o caso em que todas as novas transações referenciam de forma direta ou indireta a transação em questão (e.g., a *transação* 1 na Figura 3).

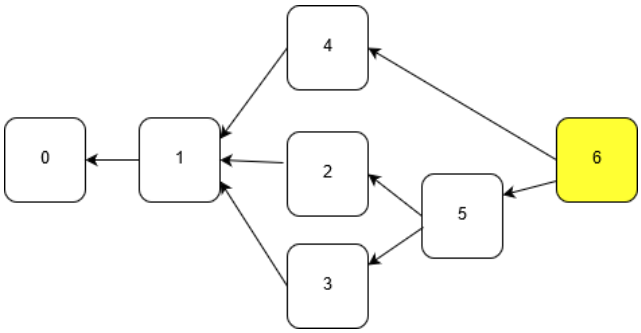


Figura 3 - Exemplo de um DAG de uma *Tangle*.

Toda vez que uma transação é referenciada, ela deixa de ser uma *ponta* e tende a ser uma transação definitiva, fazendo parte da *Tangle* e tendo seu *peso cumulativo* atualizado conforme outras transações a referenciam. Para que uma *ponta* seja selecionada como candidata a ser validada, é necessário que ela seja escolhida pelo algoritmo *Markov Chain Monte Carlo* – MCMC, cuja operação é simplificadamente descrita a seguir [Popov 2018a].

Duas *partículas*, também conhecidas como *caminhantes aleatórios* (do inglês, *random walkers*), são posicionadas em *transações* quaisquer da *Tangle* e estas passam a caminhar, de uma transação para a outra na *Tangle*, em direção às *pontas*. Esse caminho pode ser feito aleatoriamente ou tendenciosamente às transações de maior *peso cumulativo*. Na primeira forma (i.e., aleatoriamente), a próxima transação para qual a *partícula* irá transitar é selecionada aleatoriamente, portanto todas as transações têm igual probabilidade de serem selecionadas.

Já na segunda forma (i.e., tendenciosamente), cada *transação* pode possuir probabilidades distintas de ser selecionado. Suponha que uma *partícula* esteja posicionada em

uma determinada transação *x* e que caminhe com maior probabilidade em direção às transações com maior *peso cumulativo*. Assuma, por exemplo, que certa transação arbitrária *y* referencia diretamente *x*, ou seja, $y \rightarrow x$. Então a probabilidade de que essa *partícula* transite de *x* para *y* é igual a P_{xy}, sendo calculada pela Equação 1 [Popov 2018a]. Veja que as *pontas* selecionadas pelas *partículas* são então candidatas a serem referenciadas pelas novas transações que chegam ao sistema e, possivelmente, são então validadas na sequência.

$$P_{xy} = \left(e^{\left(-\alpha(H_x - H_y)\right)}\right)\left(\sum_{z:z\rightarrow x} e^{\left(-\alpha(H_x - H_z)\right)}\right)^{-1} \qquad (1)$$

Onde: o somatório representa o número total de casos possíveis, isto é, todas as transações *z* que referenciam diretamente *x*, incluindo *y*; e α é um parâmetro maior que zero, sendo diretamente proporcional ao quão tendenciosa (i.e., maior probabilidade) a seleção é em direção às transações com maior *peso cumulativo*. Por fim, H_x, H_y e H_z são, respectivamente, os *pesos cumulativos* das transações *x*, *y* e *z*.

A Figura 4 ilustra essa situação. O ponto preto na *transação x* representa a partícula responsável por selecionar qual transação será candidata a ser referenciada. Neste exemplo, a linha tracejada representa a situação em que a partícula seleciona a transação *y*.

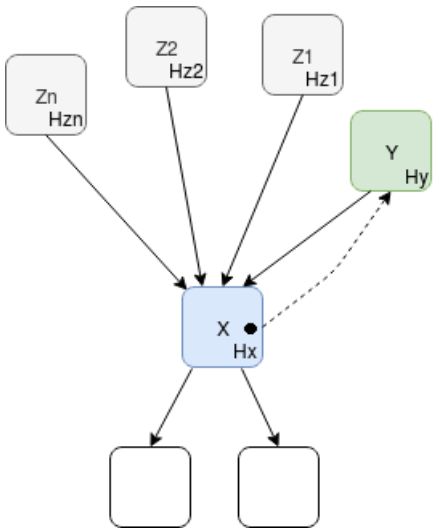


Figura 4. Seleção de uma ponta.

Também podem surgir no sistema transações que não seguem esse algoritmo, escolhendo referenciar transações antigas em vez de *pontas*. Tais transações que não seguem o algoritmo são denominadas de *pontas preguiçosas* (do inglês, *lazy tips*). Isso é feito para evitar realizar a *prova de trabalho*, pois transações antigas muitas vezes já tiveram suas *provas*

de trabalho realizadas. Entretanto, note que uma *ponta preguiçosa* possui menor probabilidade de ser selecionada ao considerar o emprego do algoritmo MCMC, quando o caminho das partículas é feito tendenciosamente às transações de maior *peso cumulativo*. Isso porque seu *peso cumulativo* é pequeno comparativamente àquele das demais transações que também referenciam a mesma transação antiga. Por exemplo, uma *ponta preguiçosa* que referenciasse a *transação* 1 da Figura 3 estaria em desvantagem, pois seu *peso cumulativo* seria igual a 1, enquanto as *transações* 2 e 3 possuem *peso cumulativo* igual a 3.

Por fim, retornando-se à Figura 2, tem-se a atividade *Divulgar Transação*. Esta atividade nada mais é que, teoricamente, atualizar a *Tangle* armazenada em todos os *nós processadores* da Rede.

4.3. ATIVIDADES DO FORNECEDOR

Com base na Figura 5, percebem-se as atividades a serem desencadeadas pelo Fornecedor. A primeira atividade refere-se a *Escutar Transação*. O Fornecedor observa periodicamente sua própria *carteira* e verifica se recebe uma transferência proveniente de um endereço pertencente ao Cliente.

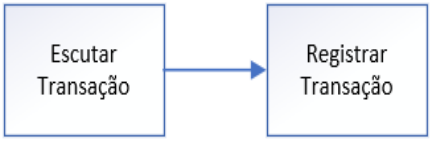


Figura 5. Atividades do Fornecedor.

A transação pode aparecer tanto para o Fornecedor quanto para o Cliente com o *status* de pendente. Isso significa que ela ainda não foi confirmada pelo sistema (i.e., disseminada pela Rede). Nessa condição, seu valor ainda não aparece disponível para o Fornecedor.

Caso a *transação* permaneça pendente por um longo período, três ações podem ser feitas para reduzir o tempo de espera. A primeira é *recolocar* (do inglês, *reattach*) a *transação*, que consiste em anexar a *transação* em uma parte diferente da *Tangle*. Novas *pontas* são então requisitadas para referenciá-la e a *prova de trabalho* é realizada novamente. A segunda é *retransmitir* (do inglês, *rebroadcast*) a *transação*, que se traduz em transmitir a mesma *transação* para um *nó processador* diferente, já que o nó processador ao qual ela havia sido submetida anteriormente poderia estar *offline* ou sobrecarregado. Por fim, é possível *promover*

a *transação*. Neste caso são criadas *transações* com valor zero que vão referenciar à *transação* com o intuito de aumentar seu peso cumulativo e, por conseguinte, aumentar suas chances de ser selecionada.

Por fim, segundo a Figura 5, tem-se a atividade *Registrar Transação*. Esta atividade não pertence de fato ao sistema IOTA, tendo sido aqui considerada para efeito de completeza da descrição de todo o processo em uma situação real. Nesta atividade estão incluídas, por exemplo, as medidas logísticas para que o produto seja entregue ao Cliente.

5. ANÁLISE DE SEGURANÇA

Uma fraude de gasto duplicado é caracterizada da seguinte forma. Logo após receber um produto (ou um serviço) de um Fornecedor, um *fraudador* submete uma nova transação que é responsável por transferir as mesmas moedas, originalmente transferidas para o Fornecedor, para um novo endereço pertencente ao próprio *fraudador*. Dessa forma, o *fraudador* invalida a transação original que transfere o montante em questão para a *carteira* do Fornecedor, mas mantém a posse do produto [Popov 2018a]. Considerando esse cenário, esta seção tem como meta avaliar o nível de segurança do sistema IOTA. Para tanto, a análise é feita sobre a sua DLT, denominada de *Tangle*.

5.1 MODELAGEM MATEMÁTICA

Admita o emprego do algoritmo MCMC, quando o caminho das partículas é feito tendenciosamente às transações de maior *peso cumulativo*, e que um *fraudador* introduz uma transação fraudulenta na *Tangle* logo após a inserção de uma transação honesta. Essa transação fraudulenta referencia outra transação mais antiga da *Tangle*. Nesse aspecto, seu comportamento se aproxima ao de uma *ponta preguiçosa*.

O *fraudador* então gera diversas novas transações que referenciam a transação fraudulenta, originando uma espécie de *ramo* (do inglês, *branch*) a partir da transação fraudulenta. Isso é realizado com a intenção de contornar o problema enfrentado pelas *pontas preguiçosas*, i.e., baixos *pesos cumulativos* que, por sua vez, resultam em uma baixa probabilidade de que elas venham a ser selecionadas. De forma contrária, ocorre então o aumento contínuo do *peso cumulativo* da transação fraudulenta, aumentando-se as chances de sucesso de fraude.

Na Figura 6 é possível visualizar esse ataque. Nela temos um exemplo de como um *fraudador* atacaria a *Tangle* ilustrada na Figura 3. A transação 6, cuja cor é verde, representa

Brazilian Journal of Development

a *transação* honesta, que efetivamente transfere o valor da mercadoria para o Fornecedor. Já a transação *F* representa a *transação* fraudulenta, que realiza o gasto duplo. Esta transação referencia a transação 3, que é uma *transação* a princípio honesta e pertencente à *Tangle*. As demais transações que referenciam *F*, de cor lilás, se referem às transações responsáveis por aumentar o *peso cumulativo* de *F*, constituindo uma espécie de *ramo* na *Tangle*.

O Fornecedor deve esperar um certo tempo até que a fraude não possa mais ocorrer, podendo então enviar o produto com segurança. Ou seja, o Fornecedor deve esperar até que a probabilidade de uma *partícula* selecionar a *transação* fraudulenta torne-se insignificante, comparativamente à probabilidade de ela selecionar transações honestas da *Tangle.*

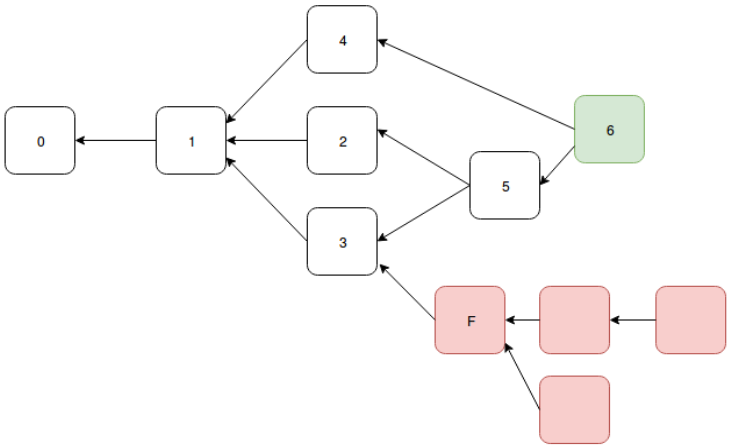


Figura 6. Ataque de gasto duplo.

Considere o cenário em que uma *transação y* fraudulenta, responsável por realizar o gasto duplo, referencia uma *transação* antiga qualquer, aqui identificada como *transação x*, que está contida na *Tangle*. O *peso cumulativo* da *transação y*, denotado por H_y, é o somatório dos pesos individuais de todas as transações que a referenciam, de forma direta ou indireta durante um intervalo de tempo Δt, somado ao seu próprio *peso*. Neste caso, todas as *transações* geradas pelo fraudador chegam ao denominado *ramo parasita*, i.e., *ramo* responsável por aumentar o *peso cumulativo* de *y*. A Equação 2 realiza esse cômputo. Com o intuito de simplificar os cálculos, é considerado que todas as transações geradas possuem *peso* igual a 1.

$$H_y = H_{oy} + \Delta t \cdot \lambda_{parasita} \qquad (2)$$

Onde: H_{oy} é o peso da transação *y*, $\lambda_{parasita}$ é a taxa de transações que chegam ao ramo parasita, e Δt contabiliza o tempo passado a partir do momento em que a transação fraudulenta foi introduzida na *Tangle*.

Por sua vez, o *peso cumulativo* de *x*, denotado por H_x, é, de maneira análoga, dado pelo somatório dos pesos de todas as transações que referenciam, direta ou indiretamente, a transação *x*, incluindo a própria transação *y* fraudulenta e todas as transações pertencentes ao *ramo* parasita. Como *x* é uma transação antiga, admita que a mesma é referenciada por todas as novas transações que chegam à *Tangle*, semelhante ao que ocorre com a transação 1 na Figura 3. Note que as transações à *Tangle* não incluem as transações que chegam ao ramo parasita. A Equação 3 realiza esse cômputo.

$$H_x = H_{ox} + \Delta t \cdot \lambda_{tangle} + H_y \quad (3)$$

Por sua vez, substituindo-se a Equação 2 na Equação 3, tem-se a Equação 4. Em seguida, substituindo-se as Equações 4 e 2 na Equação 1, obtém-se então a Equação 5, mostrada a seguir.

$$H_x = H_{ox} + \Delta t \cdot \lambda_{tangle} + H_{oy} + \Delta t \cdot \lambda_{parasita} \quad (4)$$

$$P_{xy} = \frac{e^{\left(-\alpha\left((H_{ox}+\Delta t\cdot\lambda_{tangle}+H_{oy}+\Delta t\cdot\lambda_{parasita})-(H_{oy}+\Delta t\cdot\lambda_{parasita})\right)\right)}}{\sum_{z:z\to x} e^{\left(-\alpha\left((H_{ox}+\Delta t\cdot\lambda_{tangle}+H_{oy}+\Delta t\cdot\lambda_{parasita}) - H_z\right)\right)}} \quad (5)$$

A Equação 5 calcula a probabilidade de que uma *partícula* transite de *x* para a transação fraudulenta *y*, permitindo que *y* (ou qualquer outra transação pertencente ao *ramo parasita*) seja selecionada e possivelmente validada, concretizando a fraude. Simplificando-se a Equação 5 tem-se finalmente a Equação 6.

$$P_{xy} = \frac{e^{\left(-\alpha\left(H_{ox}+\Delta t\cdot\lambda_{tangle}\right)\right)}}{\sum_{z:z\to x} e^{\left(-\alpha\left((H_{ox}+\Delta t\cdot\lambda_{tangle}+H_{oy}+\Delta t\cdot\lambda_{parasita})- H_z\right)\right)}} \quad (6)$$

Na Figura 7 é ilustrada a situação descrita acima. O grupo de transações em verde tem o seu peso cumulativo aumentado de acordo com a taxa de novas transações que chegam à

Brazilian Journal of Development

Tangle. Já o grupo de transações em amarelo, pertence ao fraudador, tem o seu peso cumulativo aumentado de acordo com a taxa de novas transações que chegam ao *ramo parasita*.

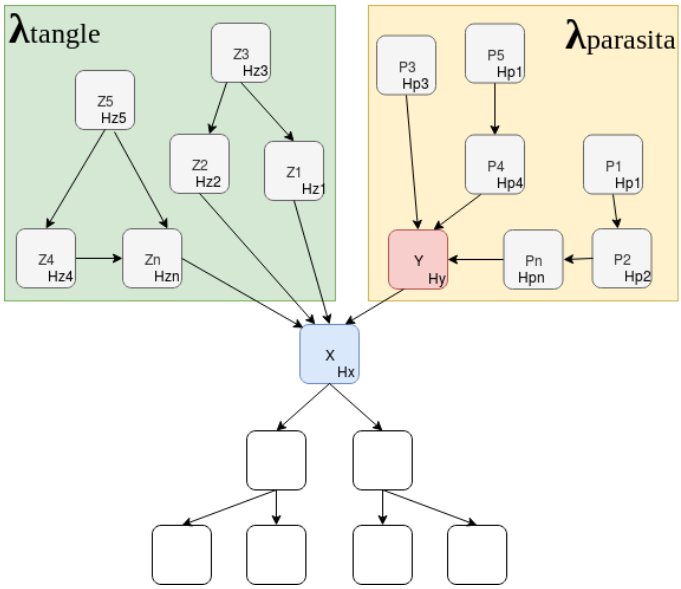


Figura 7. Ilustração da situação considerada na Equação 6.

Para uma situação em que *x* fosse referenciada apenas por *y* e outra transação *z* qualquer pertencente à *Tangle*, a Equação 6 poderia então ser reescrita na forma da Equação 7. Ou seja, diferente da Equação 6, que é utilizada para computar a probabilidade de que uma partícula transite de *x* para *y*, quando *x* é referenciada por um número qualquer de transações, a Equação 7 é utilizada apenas quando 2 transações, *y* e *z*, referenciam *x*. Sendo *y* uma transação fraudulenta referenciada pelo *ramo parasita* e *z* uma transação qualquer pertencente à *Tangle* e que, portanto, é referenciada pelas demais transações também pertencentes a ela.

$$P_{xy} = \frac{e^{(-\alpha(H_{ox}+\Delta t\cdot\lambda_{tangle}))}}{e^{\left(-\alpha\left(\left(H_{ox}+\Delta t\cdot\lambda_{tangle}+H_{oy}+\Delta t\cdot\lambda_{parasita}\right)-H_{oz}+\Delta t*\lambda_{tangle}\right)\right)}+e^{(-\alpha(H_{ox}+\Delta t\cdot\lambda_{tangle}))}} \quad (7)$$

A Figura 8 exemplifica tal situação. Neste caso, a transação *z* serve de conexão entre *x* e a *Tangle*, da mesma forma que *y* conecta *x* com o *ramo parasita*.

Brazilian Journal of Development

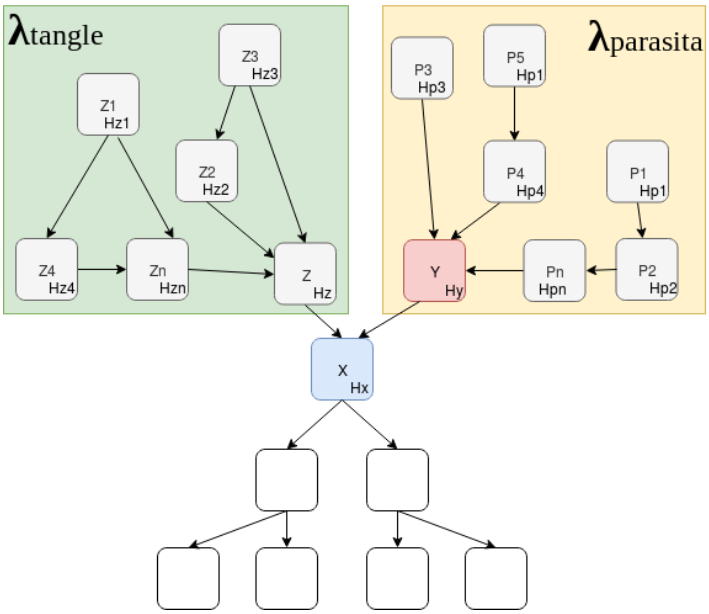


Figura 8. Influência do ramo parasita no sistema.

A ausência do termo $\lambda_{parasita}$ no numerador da Equação 7 tem a seguinte explicação. Para o cálculo de P_{xy}, apenas a diferença entre o peso cumulativo entre as transações e o valor da constante α são importantes. Portanto, ao realizar a diferença entre o peso cumulativo de *x* e de *y*, apenas o termo referente às referências feitas pelas transações na *Tangle* sobrevive, já que tanto *x* como *y* são referenciados pelo *ramo parasita*.

5.1.2 Resultados da modelagem matemática

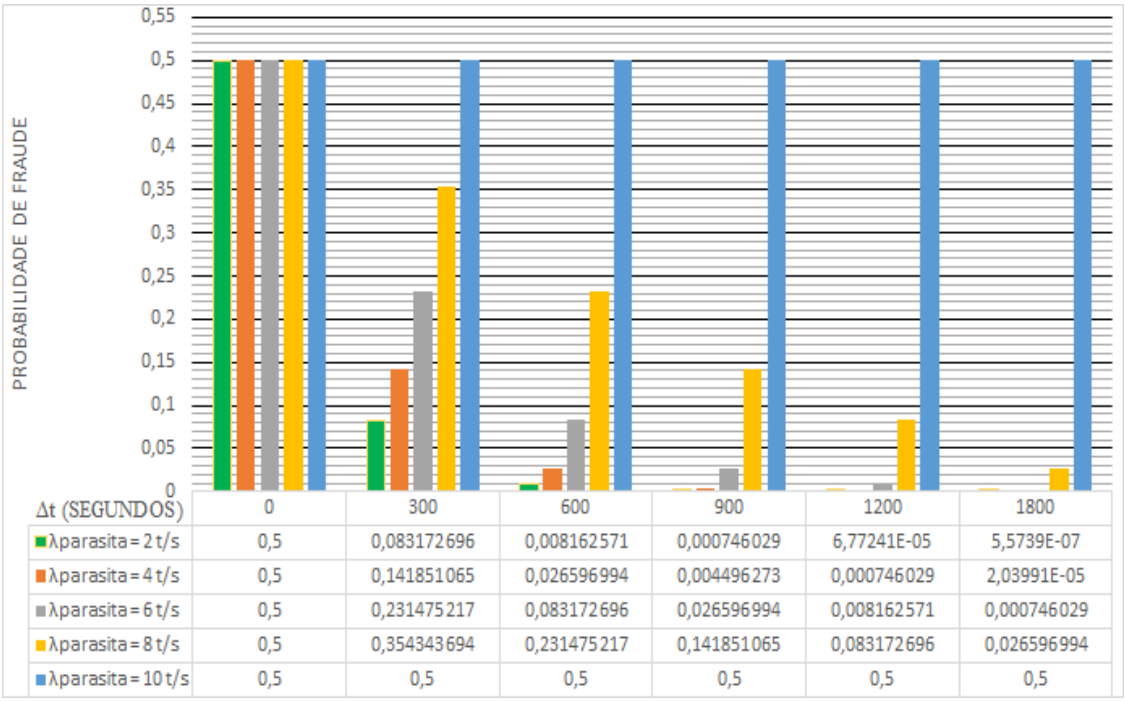
Nesta seção realizamos três experimentos cujos valores numéricos são obtidos a partir da Equação 7. O primeiro calcula a probabilidade de fraude considerando a variação do parâmetro Δt, o qual reflete a passagem de tempo desde a inserção da transação fraudulenta no sistema. O segundo experimento avalia a influência do parâmetro α nessa mesma probabilidade, onde o parâmetro α reflete o quão tendenciosa a seleção é em direção às transações com maior *peso cumulativo*. Por fim, o terceiro experimento avalia o cenário em que o fraudador possui uma capacidade de gerar transações sendo maior que a quantidade de transações por unidade de tempo recebida pela *Tangle*. Para efeito de simplificação, consideraremos a probabilidade de fraude como sendo igual à probabilidade de transição para o *ramo parasita*.

Brazilian Journal of Development

Para o primeiro experimento, tem-se o seguinte. Tendo-se em vista que o tempo de validação de uma transação no sistema IOTA é em média de 2 minutos e que o número de transações por segundo validadas neste sistema é de 10 [TangleMonitor 2019; TheTangle 2019], consideramos Δt variando desde zero até 20 min e, ainda, no ponto específico Δt igual a 30 min. Este intervalo foi assim dimensionado por ser suficiente para avaliar a probabilidade de fraude, conforme demonstrado nos resultados a serem apresentados. Ainda, para este experimento, admite-se $\alpha = 0{,}001$, e os *pesos cumulativos* iniciais de *y* e *z* iguais a 1, e o peso cumulativo de *x* igual a 3 (i.e., soma dos pesos *y*, *z* e *x*), os quais são valores usuais em experimentos de avaliação da Tangle [Gal 2018]. Por fim, consideram-se ainda os seguintes cenários situações: para λ_{tangle} igual a 10 t/s (i.e., transações por segundo), faz-se $\lambda_{parasita}$ igual a 2 t/s, 4 t/s, 6 t/s, 8 t/s e 10 (Figura 9); para λ_{tangle} igual a 100 t/s, faz-se $\lambda_{parasita}$ igual a 20 t/s, 40 t/s, 60 t/s, 80 t/s e 100 t/s (Figura 10); para λ_{tangle} igual a 1.000 t/s, faz-se $\lambda_{parasita}$ igual a 200 t/s, 400 t/s, 600 t/s, 800 t/s e 1.000 t/s (Figura 11).

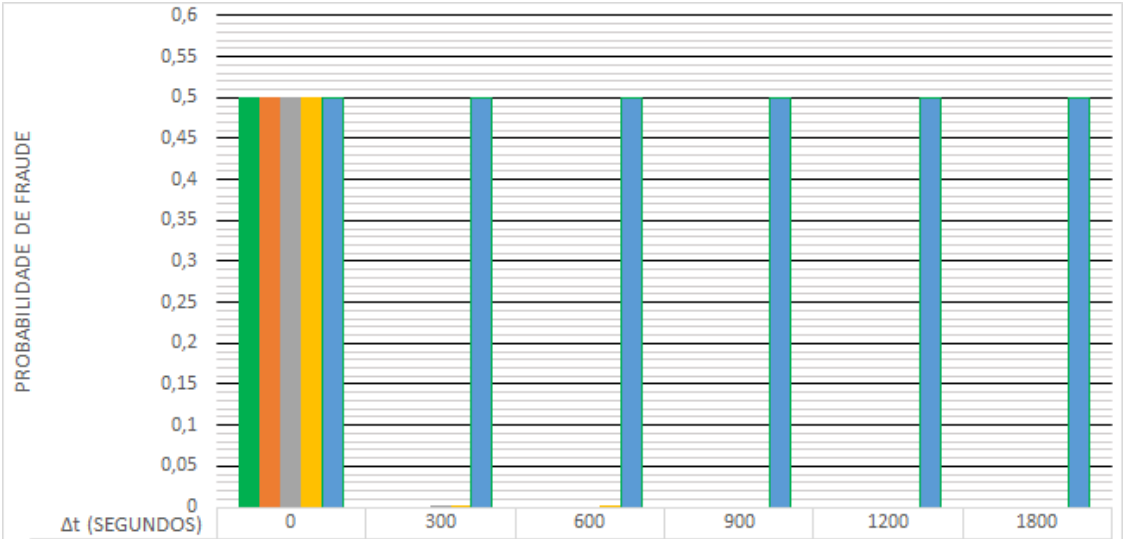
Para esse primeiro experimento, pode-se notar o seguinte. Para $\lambda_{tangle} = 10$ t/s e $\lambda_{parasita} = 4$ t/s (Figura 9), tem-se as seguintes probabilidades de fraude: 0,5 ($\Delta t = 0$ s); 0,14 ($\Delta t = 300$ s); 0,027 ($\Delta t = 600$ s); 0,004 ($\Delta t = 900$ s); 0,0007 ($\Delta t = 1200$ s); 0,0203E(-03) ($\Delta t = 1800$ s). Ou seja, observa-se que a probabilidade de fraude está intrinsecamente ligada à duração de Δt. Quanto maior este valor, menor é a probabilidade de fraude, desde que $\lambda_{tangle} > \lambda_{parasita}$. Essa mesma observação pode ser constatada para os demais valores de λ_{tangle} e $\lambda_{parasita}$, conforme ilustrado nas Figuras 10 e 11. Porém, a probabilidade de fraude se mantém constante na situação específica de $\lambda_{tangle} = \lambda_{parasita}$.

Brazilian Journal of Development



Δt (SEGUNDOS)	0	300	600	900	1200	1800
λparasita = 2 t/s	0,5	0,083172696	0,008162571	0,000746029	6,77241E-05	5,5739E-07
λparasita = 4 t/s	0,5	0,141851065	0,026596994	0,004496273	0,000746029	2,03991E-05
λparasita = 6 t/s	0,5	0,231475217	0,083172696	0,026596994	0,008162571	0,000746029
λparasita = 8 t/s	0,5	0,354343694	0,231475217	0,141851065	0,083172696	0,026596994
λparasita = 10 t/s	0,5	0,5	0,5	0,5	0,5	0,5

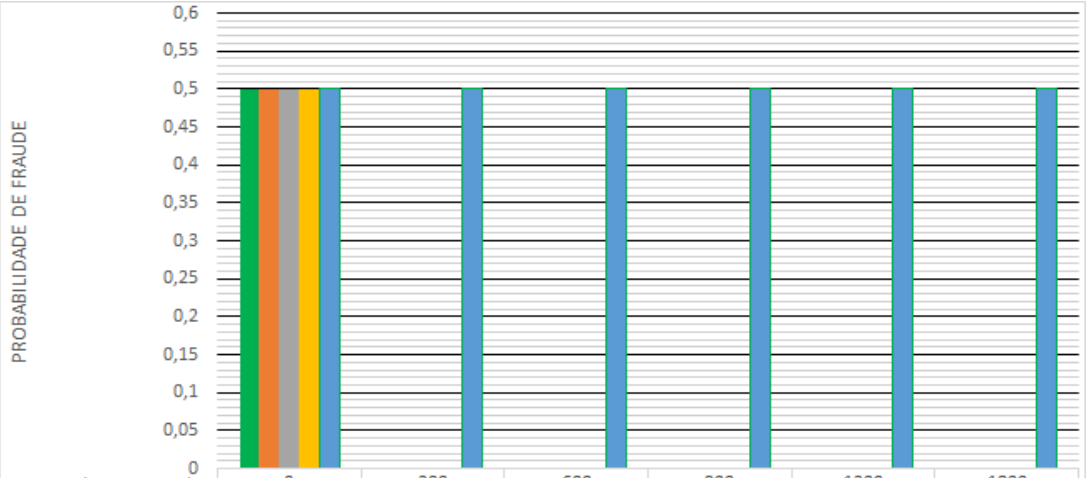
Figura 9. Probabilidade de fraude para λ_{tangle} = 10 transações por segundo.



Δt (SEGUNDOS)	0	300	600	900	1200	1800
λparasita = 20 t/s	0,5	3,77513E-11	1,42516E-21	5,38019E-32	2,03109E-42	2,89464E-63
λparasita = 40 t/s	0,5	1,523E-08	2,31952E-16	3,53263E-24	5,38019E-32	1,24795E-47
λparasita = 60 t/s	0,5	6,14417E-06	3,77513E-11	2,31952E-16	1,42516E-21	5,38019E-32
λparasita = 80 t/s	0,5	0,002472623	6,14417E-06	1,523E-08	3,77513E-11	2,31952E-16
λparasita = 100 t/s	0,5	0,5	0,5	0,5	0,5	0,5

Figura 10. probabilidade de fraude para λ_{tangle} = 100 transações por segundo.

Brazilian Journal of Development



Δt (SEGUNDOS)	0	300	600	900	1200	1800
λparasita = 200 t/s	0,5	5,8793E-105	3,4566E-209	0	0	0
λparasita = 400 t/s	0,5	6,71418E-79	4,508E-157	0	0	0
λparasita = 600 t/s	0,5	7,66765E-53	5,8793E-105	0	0	0
λparasita = 800 t/s	0,5	8,75651E-27	7,66765E-53	0	0	0
λparasita = 1000 t/s	0,5	0,5	0,5	0,5	0,5	0,5

Figura 11. probabilidade de fraude para λ_{tangle} = 1.000 transações por segundo.

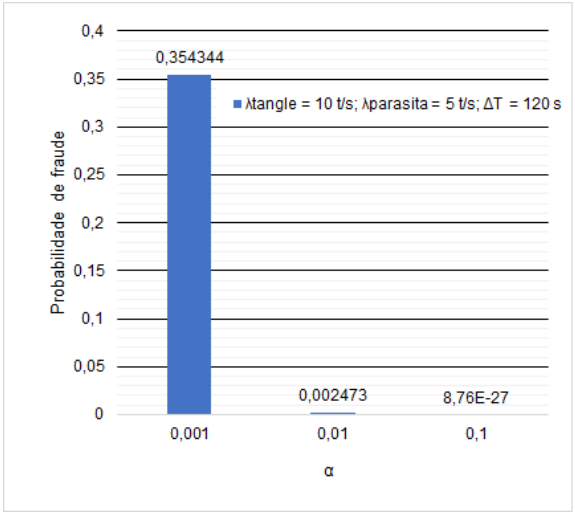


Figura 12. Probabilidade de fraude para λ_{tangle} = 10 t/s, $\lambda_{parasita}$ = 5 t/s, e Δt = 120 s.

Para o segundo experimento (Figura 12), tem-se o seguinte. Mantêm-se constantes os seguintes valores: Δt = 120 segundos, λ_{tangle} = 10 t/s, e $\lambda_{parasita}$= 5 t/s, e varia-se o valor de α = 0,001; 0,01; 0,1. Os valores arbitrados para Δt, λ_{tangle} e α neste experimento são observados no sistema real [TangleMonitor 2019; TheTangle 2019], e o valor de $\lambda_{parasita}$= 5 t/s reflete uma situação hipotética em que o fraudador possui capacidade de geração de transações igual à metade da quantidade de transações que chegam à *Tangle* por unidade de tempo, ou seja, metade do poder de processamento da rede estaria comprometido. A partir da Figura 12, pode-se notar que a probabilidade de fraude é menor quando α = 0,1.

Brazilian Journal of Development

Entretanto, vale ressaltar que, conforme o valor de α cresce, o número de *pontas permanentes* também aumenta, pois há uma maior preferência de que o sistema escolha transações antigas (ou já validadas), i.e., transações com maior *peso cumulativo* com relação às mais recentes, em detrimento das *pontas* [Kúsmierz e Gal 2018]. Assim, simplesmente aumentar o valor de α indiscriminadamente pode levar à instabilidade do sistema, como explicado a seguir. Com a falta de validação dessas *pontas*, é esperado que haja uma redução no número de usuários do sistema, já que as suas transações não estariam sendo aprovadas. Isso implicaria em também reduzir o número transações que chegam ao sistema e, consequentemente, comprometeria a segurança, já que um fraudador poderia, com maior facilidade, gerar uma quantidade de transações maior que a quantidade de transações por unidade de tempo recebida pela *Tangle*. Este cenário é melhor explorado no terceiro experimento.

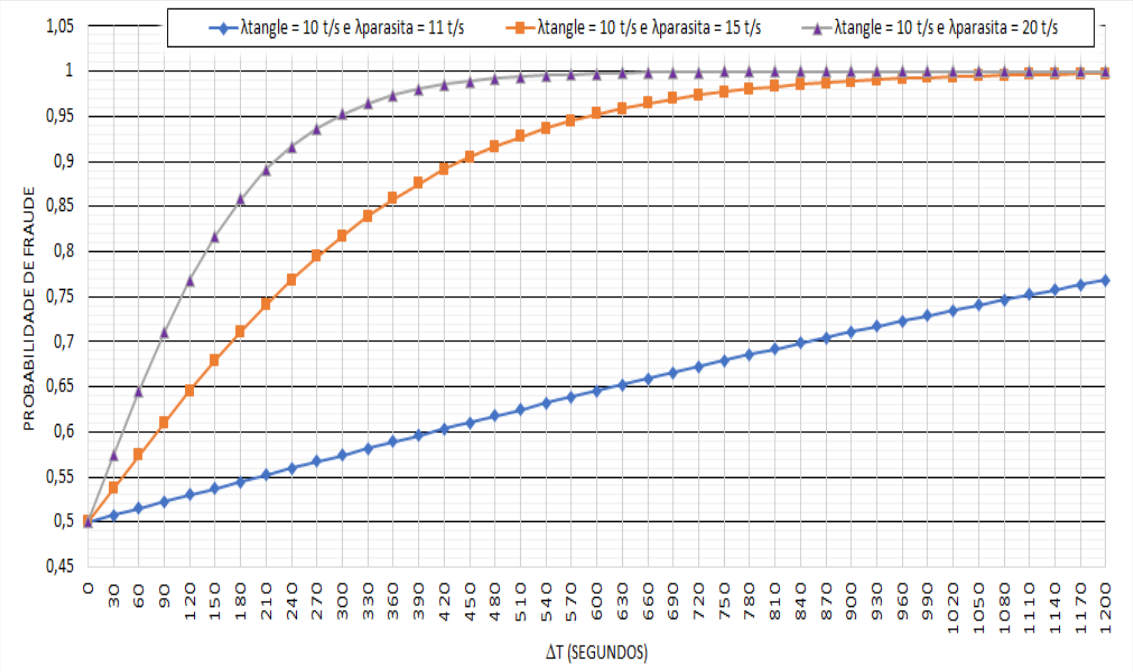


Figura 13. Probabilidade de transição para α = 0,001, sendo $\lambda_{parasita} > \lambda_{tangle}$.

Para o terceiro experimento, tem-se o seguinte. O parâmetro λ_{tangle} é mantido igual a 10, o parâmetro $\lambda_{parasita}$ é feito igual a 11, 15 e 20, Δt é variado de 0 até 1200 s, e α = 0,001. Note que os valores de $\lambda_{parasita}$ são escolhidos de modo que sejam iguais a, respectivamente, 110% (pequena vantagem), 150% (média vantagem) e 200% (grande vantagem) da quantidade média de transações recebidas por unidade de tempo pela rede. Em acordo com os resultados observados (Figura 13), é possível concluir que, conforme a diferença entre $\lambda_{parasita}$ e λ_{tangle}

aumenta, sendo $\lambda_{tangle} < \lambda_{parasita}$, a probabilidade de fraude cresce rapidamente. Além disso, há um aumento constante da probabilidade de a partícula transitar de *x* para *y* e, consequentemente, de ocorrer a fraude conforme Δt aumenta.

Em síntese, pode-se concluir que: contanto que a Rede receba mais transações por unidade de tempo que o fraudador é capaz de gerar, a probabilidade de fraude tente a zero conforme o intervalo de tempo Δt cresce; o valor de Δt necessário para obter uma dada probabilidade de fraude decresce conforme a diferença entre λ_{tangle} e $\lambda_{parasita}$ cresce. Ou seja, a probabilidade de fraude não depende propriamente dos valores dessas taxas, e sim da diferença entre elas; se $\lambda_{tangle} > \lambda_{parasita}$, a probabilidade de fraude diminui conforme o intervalo de tempo Δt cresce; se $\lambda_{tangle} < \lambda_{parasita}$, a probabilidade de sucesso de fraude aumenta conforme o intervalo de tempo Δt cresce; caso $\lambda_{tangle} = \lambda_{parasita}$, a probabilidade se mantém constante.

Note que Δt é contabilizado a partir da inserção de uma transação fraudulenta no sistema, sendo que o fraudador insere essa transação logo após a transação honesta. É possível então estabelecer uma relação entre Δt e o tempo que o cliente tem de esperar até que sua transação honesta seja aceita pelo sistema, não sendo mais possível revertê-la. Note então que esse tempo de espera é então limitado inferiormente pelo valor absoluto de Δt. Por exemplo, se $\Delta t = 600$ s para uma probabilidade de fraude máxima aceitável, então o tempo de espera do cliente deve ser maior que 600 s.

É importante destacar que, comparativamente ao conhecido e popular sistema Bitcoin, para um tempo médio de espera de 3600 s e considerando a capacidade do fraudador em 25% da capacidade da rede, a probabilidade de fraude estimada resulta em 0,05 [Rodrigues 2017]. Em cenário semelhante, para um tempo médio de espera de apenas 600 s, considerando a capacidade do fraudador em 40% ($\lambda_{parasita}$ = 4 t/s, λ_{tangle} =10 t/s, vide Figura 9) da capacidade da rede, a probabilidade de fraude seria de 0,03. Veja que a probabilidade de fraude do sistema Bitcoin é maior que no caso do sistema IOTA. Dessa forma, fica então evidente a maior robustez do sistema IOTA.

5.2 SIMULAÇÃO

A simulação é executada por meio da ferramenta Tangram-II [de Souza e Silva, Figueiredo e Leão 2009]. Esta ferramenta foi desenvolvida pela Universidade Federal do Rio de Janeiro com a participação da Universidade da Califórnia em Los Angeles. O Tangram-II permite a análise de desempenho por meio da modelagem de processos, sendo possível

resolvê-los tanto analiticamente como por meio de simulações. Os modelos de simulações são implementados a partir da definição de objetos que interagem por troca de mensagens.

O modelo de simulação desenvolvido para este trabalho é constituído de 17 objetos, usados para representar um DAG da *Tangle* elaborada para este experimento. São eles: *honesta*, *parasita*, *poisson_source*, *nó* 0, *nó* 1, ..., *nó* 13. O objeto *honesta* representa a transação honesta. O objeto *parasita* representa o *ramo parasita* que referencia à *Tangle* e que possui a transação fraudulenta. Os objetos *nó* 0, ..., *nó* 13 representam as demais transações da *Tangle*. O objeto *poisson_source* é responsável por gerar as *partículas* a uma taxa de 10 partículas por segundo, que corresponde à taxa média de um sistema real [TangleMonitor 2019; TheTangle 2019]. As partículas caminham pela *Tangle* em direção às suas *pontas*, isto é, em direção aos objetos *nó* 9, *nó* 11, *nó* 13, *honesta* e *parasita*. As relações entre esses objetos podem ser observadas na Figura 14.

Os resultados na simulação têm intervalos de confiança de 95% que estão dentro do limite de 5% dos valores estimados, tendo sido consideradas 30 execuções (rodadas) com um tempo de simulação de 10 horas cada. A mesma *Tangle* foi avaliada em três momentos distintos. O primeiro momento refere-se ao instante $t = 0$ s. Esse momento representa o estado logo após a inserção do *ramo parasita*. Isso significa que a simulação é executada com as pontas sem pesos cumulativos. O segundo momento refere-se ao instante $t = 600$ s, quando então a simulação é novamente executada. Desta vez as pontas possuem pesos cumulativos, sendo estes calculados como: $\lambda_{parasita}*600$ e $\lambda_{tangle}*600$. Por fim, no terceiro momento, para $t = 1200$ s, a simulação é novamente executada, tendo os pesos cumulativos atualizados de forma análoga ao raciocínio anterior.

De maneira geral, o modelo de simulação elaborado busca assemelhar-se ao funcionamento real do algoritmo MCMC. Mais especificamente, as *partículas* são geradas pelo objeto *poisson_source*, sendo transmitidas para o nó 0 (transação *gênesis*) e então caminham em direção às *pontas*. Um registro é então feito cada vez que uma *partícula* chega a uma *ponta*. Isso é feito com a finalidade de avaliar a proporção com que a transação *honesta* e o *ramo parasita* são selecionados pelas partículas. Essa proporção é então usada para a estimativa da probabilidade de fraude.

Brazilian Journal of Development

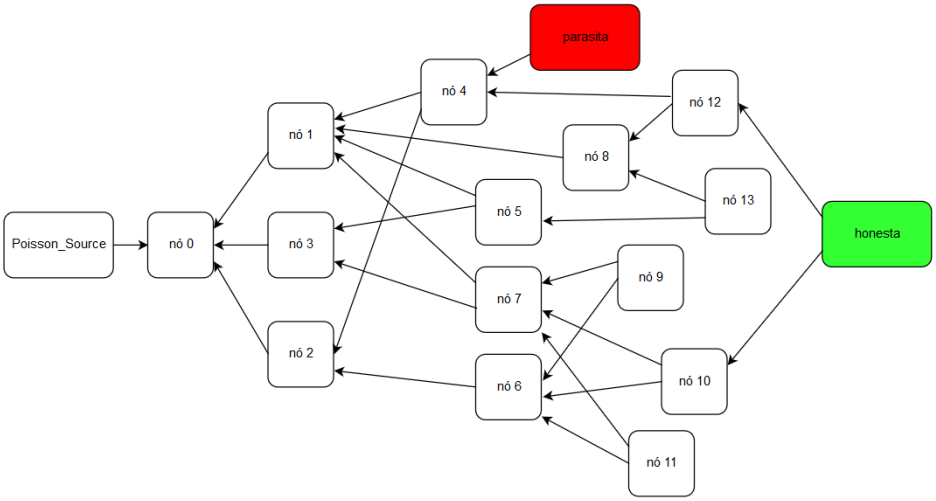


Figura 14. Relações entre os objetos.

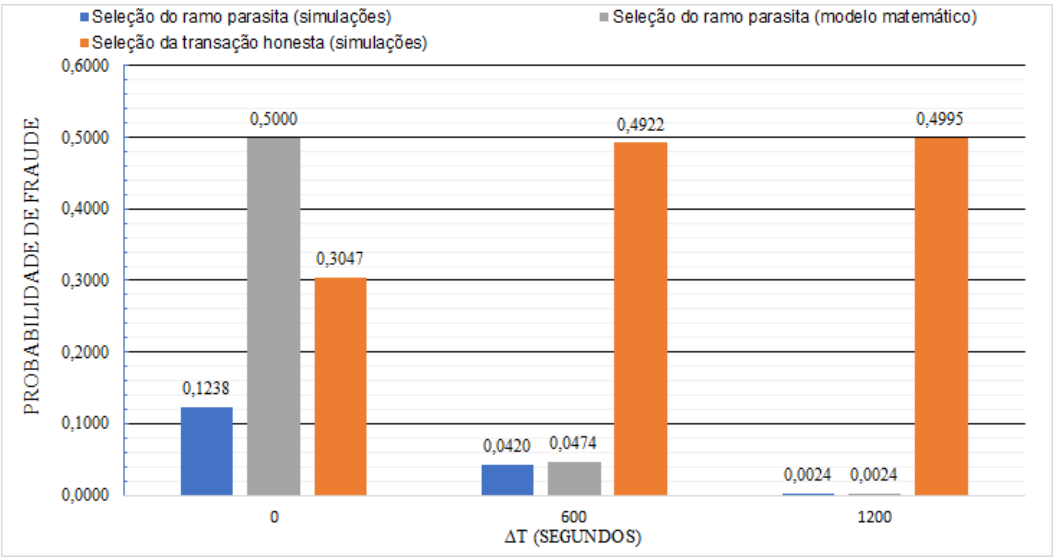


Figura 15. Valores médios das simulações.

A Figura 15 traz os resultados obtidos na simulação e a comparação com os resultados obtidos pelo modelo matemático da seção anterior. No cenário explorado na simulação, λ_{tangle} é feito igual 10/s, tendo sido escolhido a partir de observações feitas no sistema real [TangleMonitor 2019; TheTangle 2019], e tem-se $\lambda_{parasita} = \lambda_{tangle}/2$. Neste caso, como comentado na seção anterior, metade do poder de processamento da rede estaria comprometido. O valor da constante α é feito igual a 0,001, conforme discussão da seção anterior. A partir dos dados obtidos, é possível observar que, para $\Delta t = 1200$ s, a probabilidade de fraude é desprezível comparada à probabilidade de não haver fraude.

Deste experimento, também se nota a proximidade dos valores obtidos com aqueles calculados a partir do modelo matemático (vide Equação 6). Por exemplo, para $\Delta t = 600$ s, a probabilidade de fraude de 0,0474, enquanto que, na simulação, a probabilidade de fraude é 0,0420. Já para $\Delta t = 1200$ s, a probabilidade de fraude, obtida a partir do modelo matemático, é igual a 0,0024, e aquela probabilidade de fraude, mensurada na simulação, também é igual a 0,0024. Os valores da simulação terminam por validar a modelagem matemática anterior e vice-versa. Nesse sentido, as conclusões anteriores, obtidas a partir do modelo matemático, são agora ratificadas pelos resultados advindos da simulação. Mais importante, ratifica-se a significativa robustez do sistema IOTA e, ainda, a sua superioridade com relação ao conhecido sistema Bitcoin.

6. CONCLUSÕES FINAIS E TRABALHOS FUTUROS

Este artigo teve como principal objetivo a avaliação do nível de segurança do sistema IOTA. Para isso, inicialmente foi realizado um estudo de todas as atividades realizadas durante a execução de uma transação. Em seguida, através de modelagem matemática e simulação, foram realizados experimentos com o intuito de avaliar a segurança sistêmica, considerando inúmeros cenários em que um fraudador busca alterar uma informação já registrada na base de dados.

Os resultados da modelagem matemática demonstraram a significativa robustez do sistema. Nesse sentido, foi possível constatar que mesmo em situações de se ter o atacante controlando um significativo percentual do total de transações entrantes no sistema (por exemplo, até 40%), a probabilidade de fraude estimada ainda se manteria em patamares aceitáveis. Por sua vez, os resultados provenientes da simulação resultaram bem próximos daquelas da modelagem matemática, permitindo concluir sobre a validação das formulações propostas e vice-versa. Ademais, a partir da comparação desses resultados aqui obtidos com aqueles apresentados em trabalhos anteriores, foi possível concluir sobre a superioridade do sistema IOTA frente ao conhecido sistema Bitcoin.

Como trabalhos futuros, indicam-se os seguintes três caminhos possíveis. Primeiro, estender a análise aqui realizada para: cenários com grandes volumes de transações; cenários em que não são atribuídos pesos iguais e fixos a todas as transações entrantes no sistema; e cenários em que os fraudadores tentassem efetuar múltiplas transações fraudulentas em simultâneo. Segundo, realizar análises competitivas com outros sistemas de propósitos semelhantes como, por exemplo, o conhecido sistema Bitcoin. Por fim, realizar análises

competitivas com foco específico nas diferentes propostas de DLTs visando ao desenvolvimento de aplicações IoT.

REFERÊNCIAS

Antonopoulos, M. (2017). Mastering Bitcoin: Programming the Open Blockchain. 2nd Edition. Sebastopol, California: O´Reilly Media.

Bramas, Q. (2018) The Stability and the Security of the Tangle. Disponível em: https://hal.archives-ouvertes.fr/hal-01716111 . Acesso em: 17 de agosto de 2019.

Buchmann, J. et al. (2011). On the Security of the Winternitz One-Time Signature Scheme. In: Nitaj A., Pointcheval D. (eds) Progress in Cryptology – AFRICACRYPT 2011. AFRICACRYPT 2011. Lecture Notes in Computer Science, vol 6737. Springer, Berlin, Heidelberg.

Choudhary, T. et al. (2020). Convergence of Blockchain and IoT: An Edge Over Technologies. In: Hassanien A., Bhatnagar R., Khalifa N., Taha M. (eds) Toward Social Internet of Things (SIoT): Enabling Technologies, Architectures and Applications. Studies in Computational Intelligence, vol 846. Springer, Cham.

CoinMarketCap. (2019). Top 100 Cryptocurrencies by Market Capitalization. Disponível em: https://coinmarketcap.com/. Acesso em: 17 de agosto de 2019.

Cullen, A. et al. (2019). Distributed Ledger Technology for IoT: Parasite Chain Attacks. CoRR, abs/1904.00996. Disponível em https://arxiv.org/abs/1904.00996. Acesso em: 18 de agosto de 2019.

De Souza e Silva, E.; Figueiredo, R.; Leão, R. (2009). The TANGRAM-II integrated modeling environment for computer systems and networks. ACM SIGMETRICS Performance Evaluation Review, v. 36, n. 4, pp. 64-69.

Ehmke, C. et al. (2019). Properties of Decentralized Consensus Technology - Why not every Blockchain is a Blockchain. CoRR, abs/1907.09289. Disponível em: http://arxiv.org/abs/1907.09289. Acesso em: 17 de agosto de 2019.

Gal, A. (2018) The Tangle: An Illustrated Introduction. Disponível em: https://blog.iota.org/the-tangle-an-illustrated-introduction-4d5eae6fe8d4 . Acesso em: 17 de agosto de 2019.

IOTA. (2019). What is IOTA? Disponível em: https://www.iota.org/get-started/what-is-iota. Acesso em: 17 de agosto de 2019.

Iwamura, M. et al. (2014). Can we stabilize the price of a Cryptocurrency? Understanding the Design of Bitcoin and Its Potential to Compete with Central Bank Money. Disponível em: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2519367. Acesso em: 17 de Agosto de 2019.

Kannengießer, N et al. (2019). What Does Not Fit Can be Made to Fit! Trade-Offs in Distributed Ledger Technology Designs. In: Proceedings of the 52nd Hawaii International Conference on System Sciences, Wailea, Maui, HI, USA.

Kúsmierz, B. (2017) The first glance at the simulation of the Tangle: discrete model. Disponível em: https://assets.ctfassets.net/r1dr6vzfxhev/2ZO5XxwehymSMsgusUE6YG/f15f4571500a64b7741963df5312c7e7/The_First_Glance_of_the_Simulation_Tangle_-_Discrete_Model_v0.1.pdf . Acesso em: 17 de agosto de 2019.

Kúsmierz, B.; Gal, A. (2018). Probability of being left behind and probability of becoming permanent tip in the Tangle. Disponível em: https://assets.ctfassets.net/r1dr6vzfxhev/6FMwUH0b4WIyi6mm8oWWgY/8f1d7b30f7b652098a5e68b6634c63df/POLB-02.pdf . Acesso em: 17 de agosto de 2019.

Luther, W. J. (2015). Bitcoin and the Future of Digital Payments. Disponível em: http://dx.doi.org/10.2139/ssrn.2631314. Acesso em: 17 de Agosto de 2019.

Mata, R. Z. A; Rodrigues, C. K. S. (2019). Uma análise competitiva entre as tecnologias Tangle e Blockchain para Projetos de Aplicações IoT. Brazilian Journal of Development, Curitiba, v. 5, n. 7, pp. 7961-7979.

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Disponível em: https://bitcoin.org/bitcoin.pdf . Acesso em: 17 de Agosto de 2019.

Popov, S. (2018a). The Tangle – Version 1.4.3. Disponível em: https://iota.org/IOTA_Whitepaper.pdf. Acesso em: 17 de agosto de 2019.

Popov, S. (2018b). Local modifiers in the Tangle. Disponível em: https://assets.ctfassets.net/r1dr6vzfxhev/4p2Jh4jQzYwmQSqScgKW2G/6bd776742d48e6a44fca66845e956e8e/Local_Modifiers_in_the_Tangle.pdf . Acesso em: 17 de agosto de 2019.

Rodrigues, C. K. S. (2017). Sistema Bitcoin: uma análise da segurança das transações. iSys - Revista Brasileira de Sistemas de Informação, v. 10, n. 3, pp. 5-23.

Rodrigues, C. K. S; Da Silva, P. C. (2019). Uma Análise de Algoritmos de Consenso para Blockchain visando à Implementação de Sistemas de Informação Distribuídos Transparentes. Revista de Sistemas e Computação, Salvador, v. 9, n. 1, p. 163-188, jan./jun.

Sallal, M. F. (2018). Evaluation of Security and Performance of Clustering in the Bitcoin Network, with the Aim of Improving the Consistency of the Blockchain. Thesis (Doctor of Philosophy) – School of Computing, University of Portsmouth, Portsmouth, Hampshire, England.

Siris, V. A. et al. (2019). Interledger Approaches. IEEE Access, v. 7, pp. 89948-89966.

TangleMonitor. (2019). Live visualization and metrics of the IOTA Tangle. Disponível em: https://tanglemonitor.com . Acesso em: 18 de agosto de 2019.

TheTangle. (2019). IOTA Tangle Explorer. Disponível em: https://thetangle.org/. Acesso em: 18 de agosto de 2019.

Welz, W. (2019) Assuring authenticity in the Tangle with signatures. Disponível em: https://blog.iota.org/assuring-authenticity-in-the-tangle-with-signatures-791897d7b998. Acesso em: 17 de agosto de 2019.

Wood, G. (2015). Ethereum: A secure decentralized generalized transaction ledger. Disponível em: https://gavwood.com/paper.pdf . Acesso em: 17 de Agosto de 2019.

Yeow, K. et al. (2018). Decentralized Consensus for Edge-Centric Internet of Things: A Review, Taxonomy, and Research Issues. IEEE Access, v. 6, pp. 1513-1524.