Avaliação de desempenho de transações em protocolos *blockchain*: um estudo de caso com Ethereum

Danilo Bizarria de Oliveira[1], **Geraldo Lucas Fernandes do Amaral**[1], **Markel Pedrosa Duarte de Macedo**[1], **Matheus Ferreira Mesquita**[1], **Calebe de Paula Bianchini** [12]

[1] Centro Universitário FEI
São Bernardo do Campo – SP – Brasil

{unifdaoliveira, unifgamaral, uniemamacedo}@fei.edu.br,

{unifmmesquita, calebe}@fei.edu.br

[2]Mackenzie Presbyterian University

calebe.bianchini@mackenzie.br

Abstract. *Transaction processing systems are an essential part of online businesses, and require both performance and reliability to deliver the expected service. A blockchain tool is capable of offering scalability and trust when validating transactions. We propose an evaluation of the scalability potential of the Ethereum network, and its behavior in progressively more distributed scenarios. We highlight an observed increase of about 10× the number of transactions per second, when comparing an isolated node with a network of 16 nodes, the latter which also provided a drop of more than half in latency compared with the centralized scenario.*

Resumo. *Sistemas de processamento de transações são parte essencial de negócios online, e exigem tanto confiança quanto desempenho para entregar o serviço esperado. Uma ferramenta de blockchain é capaz de oferecer escalabilidade e confiança na validação de transações. A abordagem proposta neste trabalho é da avaliação do potencial de escalabilidade da rede Ethereum, e seu comportamento em cenários progressivamente mais distribuídos. Com destaque para um aumento observado de cerca de 10× o número de transações por segundo, quando comparados um nó isolado e uma rede de 16 nós, esta que também proporcionou uma queda de mais da metade da latência com relação ao cenário centralizado.*

1. Introdução

Sistemas de processamento de transações são parte essencial da arquitetura de muitos negócios e serviços *online*. Podemos citar, por exemplo, os sistemas relativos a transações financeiras, que exigem tanto confiança quanto desempenho para entregar o serviço esperado. Na maioria dessas arquiteturas, existe uma entidade centralizadora, a qual se encarrega de validar as transações inerentes do sistema.

Dado sua arquitetura distribuída, uma ferramenta de *blockchain* [Nakamoto 2008, Buterin 2014, Li et al. 2020] favorece a escalabilidade em relação a sistemas centralizados, enquanto seu protocolo de consenso formaliza a validação de transações. Isso possibilita haver confiança no estado do sistema, sem que se faça necessário haver confiança

mútua entre os nós da rede, ou confiança em uma entidade centralizadora, confia-se apenas na criptografia e em seu protocolo de consenso. Dessa forma, usuários de uma rede *blockchain* são capazes de realizar transações acordadas entre si, validadas através de assinaturas digitais, e persistidas globalmente através de um consenso estabelecido entre seus participantes que, uma vez concretizado, não pode ser trivialmente revertido.

Tendo em mente a diferença fundamental entre o modelo transacional de sistemas distribuídos, onde existe uma entidade centralizadora, e o modelo *blockchain*, que é descentralizado e valida transações através de consenso entre os nós, a proposta deste trabalho é medir e analisar o potencial de escalabilidade de uma rede *blockchain* levando em consideração duas métricas fundamentais de desempenho: *throughput* e latência.

Este trabalho apresenta uma avaliação de desempenho de transações em *blockchain*, realizando estudos de caso com o protocolo Ethereum. E está dividido da seguinte forma: a seção 2 apresenta diversos trabalhos relacionados a protocolos *blockchain* e a medição de desempenho; a seção 3 apresenta os fundamentos dos assuntos abordados; a seção 4 apresenta a descrição dos experimentos executados; a seção 5 apresenta e discute os resultados obtidos dos experimentos; e, por fim, a seção 6 sumariza o que foi inferido a partir dos resultados.

2. Trabalhos Relacionados

Nesta seção são apresentadas, em ordem cronológica, as publicações que serviram como base para a elaboração, definição do escopo e execução deste trabalho.

No trabalho de [Nakamoto 2008], é proposta uma solução ao problema de gasto-duplo em redes de pagamento *peer-to-peer*, sem a necessidade de uma autoridade central. Para obter tal garantia, a rede controla a ordenação das transações ao conectar linearmente seus *hashes* em uma cadeia contínua, que é confirmada por *Proof-of-Work*. Dessa forma, o autor calcula que a probabilidade de um ataque bem sucedido cai exponencialmente com o aumento da vantagem da cadeia honesta, medida pela diferença entre o seu comprimento e o comprimento de uma cadeia maliciosa.

No artigo de [Sompolinsky and Zohar 2015] é investigada a implicação de ter um valor de *throughput* muito alto para transações na segurança da Bitcoin contra ataques de gasto-duplo. Os autores abordam essa preocupação de segurança pela regra GHOST, uma modificação na maneira que o nó de Bitcoin constrói e reorganiza as cadeias de blocos. Os resultados apresentados mostram que com um *throughput* alto, invasores substancialmente mais fracos são capazes de reverter pagamentos que fizeram, mesmo depois de serem considerados aceitos pelos destinatários.

Em [Gervais et al. 2016] foi apresentada uma estrutura quantitativa para analisar as implicações de segurança e desempenho de vários consensos e parâmetros de rede de *blockchains* de *Proof of Work (PoW)*. Nessa estrutura, são utilizadas estratégias adversárias ideais para gasto-duplo e mineração egoísta, levando em consideração as restrições do mundo real, como propagação de rede, diferentes tamanhos de bloco, intervalos de geração de bloco, mecanismo de propagação de informações e o impacto de ataques de eclipse. A estrutura apresentada permite capturar implantações baseadas em PoW, além de variantes de *blockchain* de PoW que são instanciadas com parâmetros diferentes e comparar as compensações entre seu desempenho e provisões de segurança.

O artigo de [Li et al. 2020] apresenta Conflux, um sistema descentralizado de *blockchain* que processa blocos de forma otimista e concorrente, sem descartar qualquer bloco como uma bifurcação. Para isso, as relações entre os blocos são representadas por um grafo acíclico, e, partindo de uma ordem total dos blocos, é gerada deterministicamente uma ordem total das transações, que são então registradas na *blockchain*. O sistema foi avaliado em *clusters* Amazon EC2 com até 12 mil nós completos. Nessa configuração, o Conflux alcança um *throughput* de 9,6Mbps, equivalente a 3.480 transações por segundo.

No artigo de [Podgorelec et al. 2020], é abordado um método baseado em *machine learning* que busca introduzir a assinatura automatizada de transações de *blockchain*, incluindo também uma identificação personalizada de transações anômalas. Para validação desse método proposto, foi realizado experimento e analise de dados da rede publica Ethereum. Os resultados obtidos mostram uma visão promissora sobre o método proposto, contribuindo para um uso mais amigável de aplicativos que são baseados em *blockchain*.

3. Conceitos Fundamentais

3.1. *Blockchain*

Blockchain é uma tecnologia que possibilita a manutenção de um registro distribuído sobre o envio e recebimento de recursos digitais entre pares. Sua aplicação ocorre na implementação de cripto-moedas como Bitcoin [Nakamoto 2008], e Ether [Buterin 2014], por exemplo.

Ela é utilizada no controle de ativos virtuais. E, como analisado em [Gervais et al. 2016] e [Sompolinsky and Zohar 2015], se desafia a proporcionar segurança a uma base de dados num ambiente descentralizado, sem depender de confiança mútua entre seus nós, nem da presença de uma autoridade central para ditar o estado global do sistema. Trata-se de uma tecnologia de registro distribuído.

Seu funcionamento é, de modo geral, fundamentado na descrição de [Nakamoto 2008], e tem o objetivo de administrar a posse de um ativo digital, assegurando a transferência de propriedade de forma única e acordada entre os envolvidos. Tais garantias são obtidas, primeiramente, pela assinatura digital de cada transação por ambas as partes, seguida do agrupamento de transações em um bloco, que, quando finalizado, é inserido ao estado do sistema pela referência, no bloco atual, à identificação do bloco anterior na cadeia global, e identificado, também de forma única, por meio de um *hash* do seu conteúdo.

3.2. Métricas de Desempenho

Métricas de desempenho são dados numéricos, geralmente com relação ao tempo de execução, mensurados sobre um sistema, e são utilizadas na comparação entre diferentes arquiteturas, que tenham um objetivo comum.

São exemplos de métricas para comparação de desempenho:

- ***Throughput*:** O *throughput* é a quantidade de dados de entrada que são completamente processados num determinado intervalo de tempo [Roos et al. 2018].

Throughput refere-se a quantidade de dados que podem ser transferidos da origem para o destino dentro de um determinado intervalo específico de tempo. No contexto deste experimento, é a taxa de transações propagadas em toda rede por segundo.

- **Latência:** A latência é o tempo decorrido entre o envio de uma mensagem e o seu recebimento pelo destinatário [Roos et al. 2018]. No contexto deste experimento, a latência compreende o tempo decorrido desde que uma transação foi submetida até o momento em que o resultado dela está disponível para toda a rede. Uma latência alta significa que, para a realização da transação, o tempo decorrido para o seu recebimento foi grande. Portanto, uma baixa latência significa um comportamento eficiente da rede.

4. Avaliação de Desempenho

Este experimento destina-se a realizar uma análise de desempenho e escalabilidade de uma rede de *blockchain* que utiliza o protocolo Ethereum. Para esse objetivo, foi utilizada a ferramenta Hyperledger Caliper [Caliper 2020] para gerar e distribuir uma carga de transações entre os vários nós de uma rede *blockchain*.

Os testes foram segregados nos seguintes tipos:

1. **Centralizado:** Testa de maneira centralizada o agente de testes instanciando apenas o nó *master* do agente, sem distribuí-lo em múltiplos *workers*.
2. **Distribuídos:** Testa o agente de testes de maneira distribuída, fazendo comunicação entre um nó *master* e nós *workers* do agente instanciados para cada um dos nós da rede.

E são utilizados três tipos de contratos, sendo eles:

1. ***Open*:** Contrato responsável por simular uma abertura de conta no protocolo Ethereum.
2. ***Query*:** Contrato responsável por listar as contas que estão disponíveis na rede.
3. ***Transfer*:** Contrato responsável por simular a transferência de um valor fixo de uma conta para outra dentro do protocolo Ethereum.

O Caliper é utilizado para atuar sobre os nós do Ethereum, sendo o responsável pelo envio das transações, além da coleta das métricas que compõem o relatório de desempenho. O Mosquitto - ferramenta de mensageria (*message broker*) - é utilizado para mediar a comunicação entre os nós (*master* e *workers*) do Caliper, agente de testes do experimento. Ambas estas ferramentas são executadas através do Docker (ferramenta de containerização).

Foram realizadas cinco sessões de teste, onde se escalou a rede Ethereum quanto a sua quantidade de nós, em progressão geométrica de razão 2. Portanto, a primeira sessão se dá com 1 nó; a segunda, com 2; com 4 na terceira, e assim por diante, até a última sessão, com 16 nós. Em cada teste temos apenas um nó Ethereum por máquina, um *worker* do Caliper associado a este nó sendo executado em cada máquina, e uma outra máquina dedicada a executar a instância do Mosquitto e o nó *master* do Caliper. Durante o processo descrito nesta etapa, o Caliper coleta valores das métricas de *throughput*, latência e taxa de sucesso.

A Figura 1 representa um ambiente com 5 máquinas, onde 4 máquinas são pertencentes à rede Ethereum e a 5ª máquina representa o nó *master* do agente de testes.

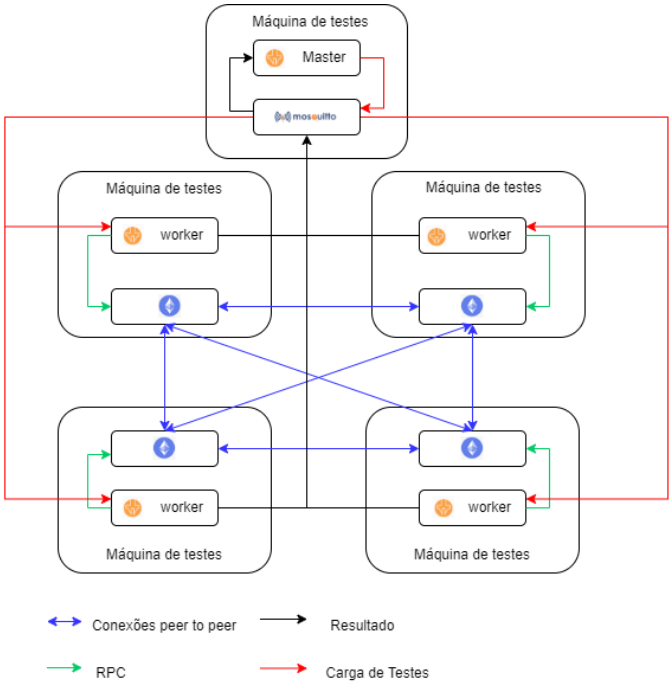

Figura 1. Diagrama do ambiente configurado para uma rede de 4 nós

Fonte: Autores

5. Resultados e Discussão

Esta seção apresenta os resultados obtidos dos experimentos detalhados na proposta experimental, envolvendo redes distribuídas, *blockchain* e contratos inteligentes. Foram feitas quatro execuções por teste descrito. A fim de gerar métricas para comparação de duas configurações do agente de testes, apenas na escala de quatro máquinas foram realizadas uma sessão de testes de maneira centralizada e outra de maneira distribuída. A seguir, são apresentados de forma agregada os resultados da realização dos testes.

5.1. Teste Centralizado

Neste experimento, para verificação do comportamento do agente de testes sobre a rede, foram realizados dois testes sobre uma rede de quatro nós Ethereum: o primeiro com um agente de testes atuando de maneira centralizada, o segundo com o agente de testes distribuído como os nós da rede sob teste. Como critérios avaliativos, os gráficos na Figura 2 foram consolidados com os resultados dos testes.

Com este teste, é possível verificar na Figura 2 que o agente de testes, atuando de maneira distribuída em conjunto com os nós da rede, contribui para o aumento da taxa média de transações processadas (*throughput*). Por outro lado, de maneira centralizada, o agente de testes foi diagnosticado como um gargalo para a taxa média de transações sucedidas, uma vez que a rede apresentava capacidade de processamento suficiente para uma maior quantidade de transações efetuadas com sucesso.

5.2. Testes Distribuídos

Nesse experimento, para avaliação do comportamento da rede, foram realizados testes de escalabilidade tanto com a rede, como com o agente de testes. Em suma, foi feita a

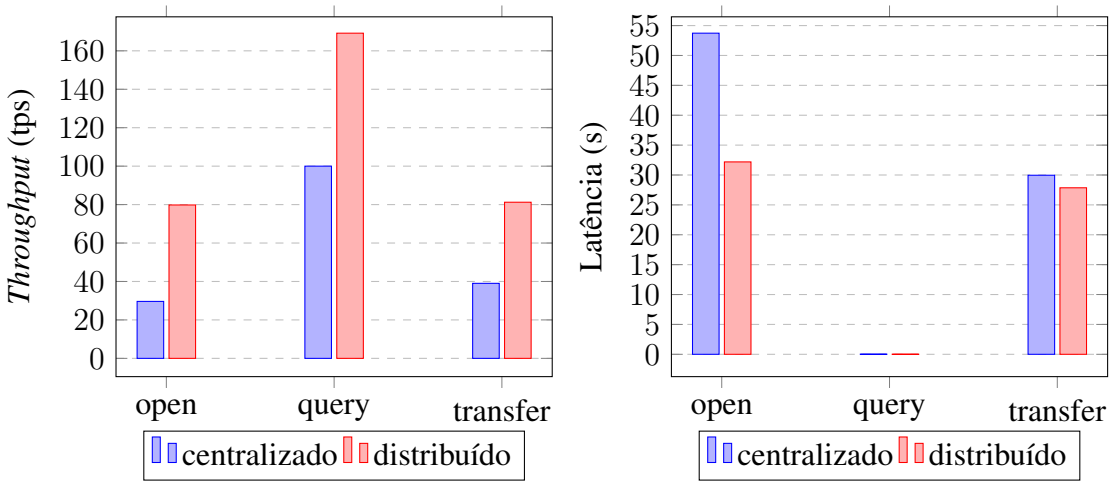

Figura 2. *Throughput* e Latência: Caliper centralizado vs. Caliper distribuído

progressão geométrica de razão 2 para demonstrar a escalabilidade da rede.

Os resultados relacionados à taxa média de transações processadas com sucesso podem ser observados na Figura 3. E observa-se que, nos testes com o contrato *open*, conforme a progressão da quantidade de máquinas foi sendo realizada, a taxa média de transações processadas com sucesso foi crescente. Nos testes de *query*, o comportamento descrito também pode ser observado, porém o teste realizado com 8 máquinas teve uma pequena queda na taxa média de transações processadas com sucesso. Após queda, o comportamento crescente se manteve. Já nos testes de *transfer* foi observado a tendência crescente da taxa média de transações processadas com sucesso conforme progressão da quantidade de máquinas, mantendo assim os padrões de escalabilidade da rede.

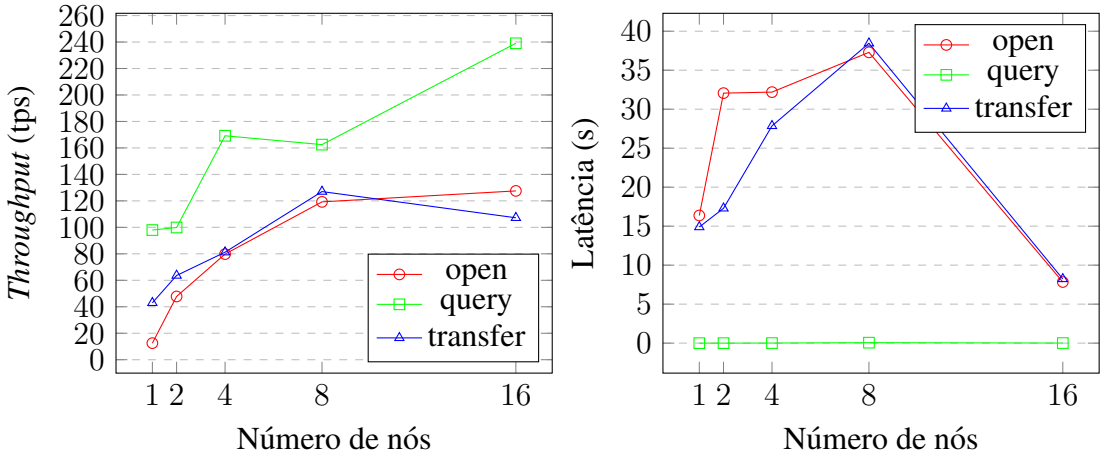

Figura 3. *Throughput* e Latência $\times$ Número de nós

Como ilustrado na Figura 3, observa-se nos testes com o contrato *open* que, conforme a progressão da quantidade de máquinas foi sendo realizada, a latência teve comportamento decrescente. Nos testes de *query*, pode ser observado que a latência se manteve estável, uma vez que esse processo não exige tanto processamento da rede. Já nos testes de *transfer*, puderam ser observados dois comportamentos: o primeiro, um comportamento crescente que pode ser observado na progressão de um até oito nós, e o segundo,

já na escala de dezesseis máquinas, no qual a latência foi reduzida de maneira significativa.

6. Conclusão

Pela estrutura descentralizada de redes *blockchain*, pode-se inferir, intuitivamente, que a escalabilidade é um atributo inerente a elas, considerando que o aumento no número de nós na rede facilita a propagação de transações. Através deste experimento, observou-se uma melhora de desempenho com a escalada da rede a mais nós, refletida no aumento de *throughput* coletado nos testes utilizando 1, 2, 4 e 8 máquinas. No teste com 16 máquinas, apesar de ter sido observada uma queda no *throughput*, em relação aos testes anteriores, esta queda é acompanhada por um queda mais acentuada na latência. Isto mostra que, apesar de haver uma perda no *throughput*, a queda mais acentuada da latência indica um aumento no desempenho geral da rede.

Para elaborar melhor este ponto, pode-se pensar no cenário observado como um *trade-off*. Analisando as transações de transferência, nos testes com 1, 2, 4 e 8 máquinas, houve um aumento gradual no *throughput*, partindo de uma média de 42.92 transações por segundo, nos testes com 1 nó, para 127.03 transações por segundo, nos testes com 8 nós, quase o triplo do *throughput* centralizado. Nestes mesmos testes, a latência média das transações de transferência tem um aumento de, aproximadamente, 2.7 vezes. Agora, observando o que acontece quando se aumenta o número de nós de 8 para 16, nota-se uma queda acentuada na latência média, que variou de 38.45 para 8.23 segundos, enquanto a queda no *throughput* foi menos expressiva, variando de 127.03 para 107.23. Isso mostra que os ganhos de desempenho com a queda de latência foram mais expressivos do que a perda de *throughput*, o que resulta em um *trade-off* positivo para o desempenho da rede.

Nas transações de abertura de conta (*open*), esta tendência é mais evidente. Nela, o *throughput* variou de 12.53 para 127.53 nos testes de 1 à 16 máquinas, sempre com ganho de *throughput* a medida que se aumentou o número de máquinas. Já a latência variou de 16.37 até 37.29 segundos nos testes de 1 à 8 máquinas, caracterizando uma perda inicial de desempenho, porém, com 16 nós, observa-se uma queda para 7.84 segundos, quase um quinto do seu valor mais alto, o que representa um ganho de desempenho com relação a todas as escalas anteriores, desde apenas 1 nó até 8 nós em funcionamento.

Observa-se, então, uma relação diretamente proporcional entre desempenho e escalabilidade na rede *blockchain* testada, o que dá indícios, assim, de uma tendência positiva entre desempenho e escala para aplicações distribuídas implementadas neste contexto.

Referências

Buterin, V. (2014). Ethereum: A next-generation smart contract and decentralized application platform. `https://ethereum.org/669c9e2e2027310b6b3cdce6e1c52962/Ethereum_Whitepaper_-_Buterin_2014.pdf`. Ethereum Organizaiton.

Caliper, P. (2020). Architecture. `https://hyperledger.github.io/caliper/v0.3.2/architecture/`. Hyperledger Caliper.

Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H., and Capkun, S. (2016). On the security and performance of proof of work blockchains. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*.

Li, C., Li, P., Zhou, D., Yang, Z., Wu, M., Yang, G., Xu, W., Long, F., and Yao, A. C.-C. (2020). A decentralized blockchain with high throughput and fast confirmation. In *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, pages 515–528. USENIX Association.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. `https://bitcoin.org/bitcoin.pdf`. Bitcoin Project.

Podgorelec, B., Turkanović, M., and Karakatič, S. (2020). A machine learning-based method for automated blockchain transaction signing including personalized anomaly detection. *Sensors (Basel, Switzerland)*, 20.

Roos, S., Moreno-Sanchez, P., Kate, A., and Goldberg, I. (2018). Settling payments fast and private: Efficient decentralized routing for path-based transactions. *Proceedings 2018 Network and Distributed System Security Symposium*.

Sompolinsky, Y. and Zohar, A. (2015). Secure high-rate transaction processing in bitcoin. In *Financial Cryptography*.