

# Análise Fatorial dos Parâmetros de Bloco e seu Impacto na Latência de Transações no Hyperledger Fabric

**Ericksulino Manoel de Araújo Moura<sup>1</sup>, Glauber Dias Gonçalves<sup>1</sup>,  
Allan Edgard Silva Freitas<sup>2</sup>, André Soares<sup>1</sup>**

<sup>1</sup>Universidade Federal do Piauí (UFPI), Teresina, Piauí, Brasil

<sup>2</sup>Instituto Federal da Bahia (IFBA), Salvador, Bahia, Brasil

{ericksulino, ggoncalves, andre.soares}@ufpi.edu.br

allan@ifba.edu.br

**Abstract.** *Blockchain technology offers robust and secure solutions for decentralized data management and is widely used in various industries. This study investigates the impact of BatchSize, BatchTimeout and Transaction Rate per Second (TPS) parameters on the performance of Hyperledger Fabric, using a 2<sup>3</sup> factorial experiment to explore how these variables influence transaction latency. The results indicate that BatchTimeout is the parameter that most affects latency, with the interaction between BatchSize and BatchTimeout also playing a significant role. These findings highlight the importance of properly tuning these parameters and understanding their interactions to optimize the performance of the Hyperledger Fabric network in high-volume transaction environments. The study emphasizes the need for precise and adaptive control of system settings to maximize transaction execution efficiency.*

**Resumo.** *A tecnologia Blockchain oferece soluções robustas e seguras para o gerenciamento descentralizado de dados, sendo amplamente utilizada em diversos setores. Este estudo investiga o impacto dos parâmetros BatchSize, BatchTimeout e Taxa de Transações por Segundo (TPS) no desempenho do Hyperledger Fabric, utilizando um experimento fatorial 2<sup>3</sup> para explorar como essas variáveis influenciam a latência das transações. Os resultados indicam que o BatchTimeout é o parâmetro que mais afeta a latência, com a interação entre BatchSize e BatchTimeout também desempenhando um papel significativo. Esses achados destacam a importância de ajustar corretamente esses parâmetros e compreender suas interações para otimizar o desempenho da rede Hyperledger Fabric em ambientes de alto volume de transações. O estudo enfatiza a necessidade de um controle preciso e adaptativo das configurações do sistema para maximizar a eficiência na execução de transações.*

## 1. Introdução

A tecnologia Blockchain é considerada uma inovação disruptiva, especialmente nos setores industrial e de serviços, ao oferecer soluções robustas para o gerenciamento de dados de forma segura e descentralizada [Xu et al. 2019]. Seu principal objetivo é facilitar o registro seguro de interações e transações entre partes distintas, como indivíduos ou organizações, que podem não compartilhar conhecimento ou confiança mútuos. Isso foi

possibilitado pela evolução e integração de tecnologias como criptografia assimétrica e protocolos de consenso distribuído via comunicação peer-to-peer, que são a essência dos blockchains [Greve et al. 2018].

As principais características do blockchain no registro de transações incluem imutabilidade, auditabilidade e consistência, derivadas de tecnologias essenciais para a criação de sistemas distribuídos, como criptografia assimétrica, algoritmos de consenso e redes peer-to-peer [Greve et al. 2018]. No entanto, melhorar o desempenho é essencial para garantir que plataformas blockchain sejam eficazes em aplicações industriais, onde transações rápidas são frequentemente necessárias. Este requisito se torna ainda mais crítico em redes públicas amplamente conhecidas, como Bitcoin e Ethereum [Sousa et al. 2021].

Neste contexto, surgem as redes blockchain permissionadas (ou privadas) [Androulaki et al. 2018], que se apresentam como uma alternativa atrativa para organizações com infraestrutura e equipe técnica próprias, buscando evitar problemas de custos (tarifação) e desempenho instável de redes públicas como Ethereum e Bitcoin [Sousa et al. 2021]. O Hyperledger Fabric é uma das plataformas mais populares para blockchain permissionada<sup>1</sup>, oferecendo recursos para a implantação de redes privadas entre organizações e o desenvolvimento de aplicações sobre essas redes.

Na plataforma Hyperledger Fabric, os participantes formam um consórcio e compartilham os custos da infraestrutura, visando maior desempenho em relação às redes públicas. A plataforma também permite a configuração de parâmetros importantes, como o “Tamanho do Lote” e o “Tempo Limite do Lote”, que influenciam o crescimento do livro-razão, além da taxa de transferência e latência [Liu et al. 2021], [Baliga et al. 2018]. Ao avaliar o desempenho de ambientes blockchain, três métricas primárias são comumente avaliadas: Taxa de transferência, Latência e Utilização de recursos [Melo et al. 2022]. A maioria dos artigos publicados sobre avaliação de desempenho de blockchain emprega pelo menos uma dessas métricas [Li et al. 2021].

Para conduzir esta pesquisa, propomos uma metodologia fatorial para avaliar o impacto das configurações de *BatchSize*, *BatchTimeout* e *TPS* (Transações Por Segundo) no desempenho do Hyperledger Fabric. Utilizamos um experimento fatorial 2<sup>3</sup>, que permite a análise das interações e efeitos dessas variáveis na latência média das transações. O planejamento experimental fatorial é uma abordagem vantajosa por permitir a análise simultânea de múltiplos fatores que impactam o desempenho [Jain 1990]. No nosso estudo, realizamos experimentos com combinações específicas de *BatchSize* (5 e 20), *BatchTimeout* (1s e 4s) e *TPS* (5 e 50). Uma latência mais baixa é altamente desejável, pois reflete um processamento de transações mais rápido, essencial para aplicações em tempo real [Androulaki et al. 2018].

Este estudo contribui para a análise dos efeitos das configurações de *BatchSize*, *BatchTimeout* e *TPS* no desempenho do Hyperledger Fabric, especificamente na latência das transações. Também visa identificar as combinações mais eficientes, destacar os parâmetros com maior impacto na latência e propor diretrizes práticas para otimizar redes Hyperledger Fabric. Os resultados indicam que o *BatchTimeout* teve o maior impacto na latência, seguido pela interação entre *BatchSize* e *BatchTimeout*. Além disso, taxas

---

<sup>1</sup><https://www.ibm.com/topics/hyperledger>

de transação mais altas (*TPS*) contribuíram para a redução da latência em determinados cenários.

## 2. Trabalhos Relacionados

A avaliação de desempenho em sistemas blockchain, especialmente no Hyperledger Fabric (HLF), tem sido amplamente pesquisada, abordando aspectos como latência, throughput e parâmetros de configuração. Cada estudo oferece uma perspectiva única, que se diferencia do nosso foco.

O estudo de [de Sá et al. 2013] propõe um ajuste dinâmico de parâmetros de agrupamento de requisições em sistemas distribuídos, focando na melhoria do desempenho do PBFT (Practical Byzantine Fault Tolerance) usando controle por feedback. Em contraste, nosso estudo se concentra especificamente no impacto das interações entre os parâmetros *BatchSize*, *BatchTimeout* e *TPS* no desempenho do Hyperledger Fabric. Shalaby et al. [Shalaby et al. 2020] conduzem experimentos para avaliar a latência e throughput do HLF, considerando o impacto do número de peers endossantes e parâmetros de configuração como o tempo limite de lote e tamanho do lote. Embora seu foco seja similar, nossa pesquisa utiliza um experimento fatorial  $2^3$  para estudar como essas variáveis interagem e afetam a latência de forma controlada, indo além de uma análise isolada de cada parâmetro.

Chacko et al. [Chacko et al. 2021] investigam falhas de transação no HLF e propõem melhorias usando ferramentas como HyperLedgerLab e otimizações recentes como Fabric++ e Streamchain. Enquanto isso, nosso estudo não se foca em falhas transacionais, mas em parâmetros de configuração que influenciam a performance em termos de latência e throughput. Saeed [Saeed et al. 2022] examina o desempenho do HLF em sistemas de e-votação, abordando diferentes tamanhos de bloco e taxas de transação. Embora sua pesquisa envolva um contexto específico, nossa abordagem é mais ampla, tratando de parâmetros essenciais para o desempenho do HLF, aplicáveis a diferentes cenários industriais.

O estudo de [Gaba et al. 2022] foca na Hyperledger Fabric Private Blockchain Network (HFPBN) e sua aplicação em Redes Veiculares Ad-hoc (VANETs), analisando o impacto do tamanho de bloco no throughput e latência. Essa aplicação específica para VANETs difere da nossa pesquisa, que se concentra em uma análise mais geral da interação de parâmetros como *BatchSize* e *BatchTimeout* em um contexto de rede privada. Wai et al. [Wai and Thein 2023] investigam como o tamanho do bloco e o tempo limite de lote impactam o desempenho do HLF, utilizando a ferramenta Hyperledger Caliper para medir throughput e latência. Em comparação, nosso estudo não apenas investiga esses parâmetros, mas também a interação entre eles, utilizando um plano experimental fatorial para entender as relações mais complexas.

Kadhm et al. [Kadhm et al. 2023] exploram o uso do HLF em e-government, com foco em segurança e desempenho sob altas cargas de transação, variando a taxa de envio e o número de organizações. Nossa estudo, por outro lado, não foca em condições de carga extrema, mas investiga como diferentes configurações de parâmetros impactam o desempenho em um cenário mais controlado e previsível. Por fim, Roy et al. [Roy and Ghosh 2024] propõem o FabMAN, um algoritmo adaptativo para otimizar o tamanho e o tempo limite de lote no HLF, visando melhorias na escalabilidade. Nossa

estudo vai além ao explorar o impacto combinado de *BatchSize*, *BatchTimeout* e *TPS*, utilizando uma abordagem factorial para investigar as interações entre esses parâmetros e seu efeito no desempenho geral.

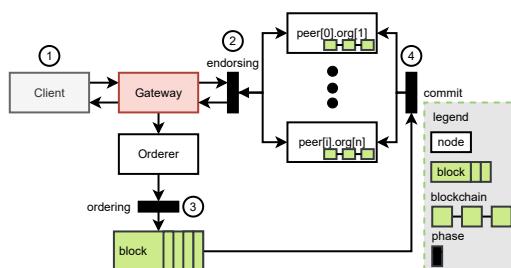
A principal contribuição do nosso estudo é a análise controlada das interações entre os parâmetros *BatchSize*, *BatchTimeout* e *TPS*, utilizando um planejamento factorial 2<sup>3</sup>, permitindo a identificação das interações mais significativas e avançando as pesquisas existentes que exploram apenas o impacto isolado ou cenários específicos de aplicação.

### 3. Hyperledger Fabric

A plataforma para redes permissionadas Hyperledger Fabric é uma das mais populares atualmente. Ela é um grande projeto de código fonte aberto envolvendo mais de 35 organizações e 200 desenvolvedores<sup>2</sup>. Hyperledger Fabric [Androulaki and et al. 2018] é uma tecnologia blockchain autorizada que é um livro-razão imutável e distribuído. A rede blockchain da plataforma Hyperledger Fabric usa a estratégia executar-ordenar-validar para processar blocos de transações. Além disso cada um dos participantes de uma rede Hyperledger Fabric possui uma cópia do livro razão. O mecanismo usado para validar as transações e criar blocos no Hyperledger Fabric é o *Practical Byzantine Fault Tolerance* (PBFT) [Castro et al. 1999].

O Hyperledger Fabric é um blockchain exclusivo plataforma projetada para casos de uso onde privacidade, escalabilidade, personalização e desempenho são crítico [Kadhm et al. 2023]. Ele é altamente escalável e capaz de processar um elevado volume de transações por segundo. Isso é conseguido através de sua abordagem única ao consenso, que permite para processamento de transações paralelas e fora da cadeia armazenamento de dados [Ma et al. 2020]. As aplicações dentro do HLF são conhecidas como *chain code* e são baseadas no conceito de contratos inteligentes definidos para redes Ethereum.

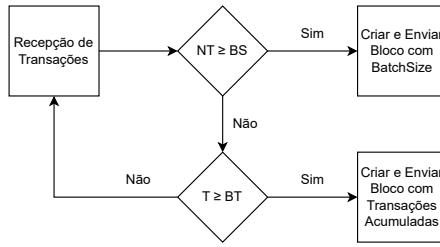
A Figura 1 demonstra o fluxo de transações na rede HLF.



**Figura 1. Hyperledger Fabric - Fluxo de transação**

O processo de agrupamento e envio de transações no Hyperledger Fabric é gerenciado pelos parâmetros *BatchSize* e *BatchTimeout*. Quando transações são recebidas, o sistema verifica se o número acumulado atinge o *BatchSize*. Se alcançado, um bloco é enviado; caso contrário, verifica-se se o tempo definido pelo *BatchTimeout* foi atingido. Se o tempo expirar, o bloco é enviado com as transações acumuladas, mesmo que o *BatchSize* não tenha sido completado. Esse processo é ilustrado na Figura 2.

<sup>2</sup>Mais informações podem ser encontradas em <https://hyperledger-fabric.readthedocs.io>.



**Figura 2. Fluxograma de Processamento de Transações no Hyperledger Fabric: Intereração entre NT (Número de Transações), BT (BatchTimeout), BS (BatchSize) e Tempo (T).**

#### 4. Metodologia de Avaliação

Esta seção descreve a metodologia usada para avaliar o desempenho do Hyperledger Fabric (HLF) em diferentes configurações de parâmetros, focando na latência média das transações. A abordagem envolve a realização de experimentos controlados para analisar os efeitos das variáveis *BatchSize* (BS), *BatchTimeout* (BT) e a taxa de transações por segundo (TPS). Os experimentos foram realizados em uma rede HLF configurada com os valores padrão de *BatchSize* igual a 10 e *BatchTimeout* de 2 segundos. Para os testes, os valores de BS, BT e TPS foram variáveis, conforme descrito abaixo:

- **BatchSize (BS):** 5 e 20.
- **BatchTimeout (BT):** 1 segundo e 4 segundos.
- **Taxa de Transações por Segundo (TPS):** 5 e 50.

Essas variações foram escolhidas para analisar como alterações nos parâmetros impactam a latência das transações, com o objetivo de observar o desempenho sob diferentes cenários. A configuração do experimento seguiu um plano fatorial  $2^3$ , resultando em 8 combinações distintas de parâmetros, como mostrado na Tabela 1. Para cada configuração, medimos a latência média das transações.

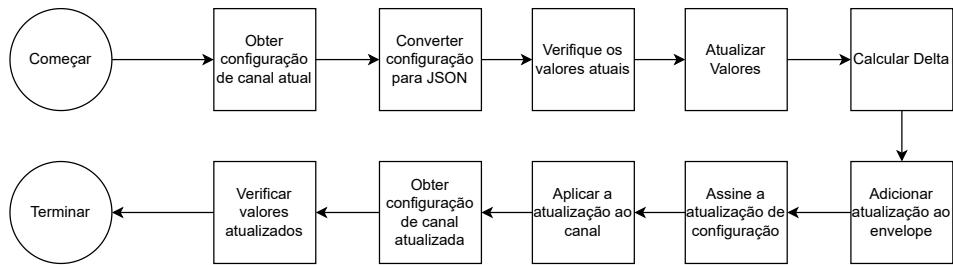
##### 4.1. Processo de Atualização dos Parâmetros

A atualização dos parâmetros de configuração no Hyperledger Fabric envolve várias etapas, descritas a seguir:

1. **Modificar os valores de *BatchSize* e *BatchTimeout* no arquivo JSON:** Os valores de *BatchSize* e *BatchTimeout* são alterados no arquivo de configuração JSON do Hyperledger Fabric, refletindo as novas configurações desejadas.
2. **Converter o arquivo JSON para o formato ProtoBuf:** O arquivo JSON modificado é convertido para o formato ProtoBuf, utilizado internamente pelo Hyperledger Fabric para processar as configurações.
3. **Calcular o delta entre a configuração original e a modificada:** O delta, ou diferença, entre a configuração original e a modificada é calculado para aplicar apenas as alterações necessárias.
4. **Adicionar o delta ao envelope de transação e assiná-lo:** O delta é adicionado ao envelope de transação e assinado, garantindo que as alterações sejam validadas e autorizadas.

5. **Aplicar a atualização ao canal e verificar as mudanças:** A atualização é aplicada ao canal e verificada para garantir que as novas configurações foram registradas corretamente na rede.

Esse processo foi automatizado com um script shell, que acelera a aplicação das configurações e minimiza erros manuais. O fluxo de atualização é ilustrado na Figura 3.



**Figura 3. Fluxo de Atualização dos Parâmetros de Configuração do Hyperledger Fabric.**

#### 4.2. Execução dos Experimentos e Ambiente de Teste

Os experimentos foram organizados com base no plano factorial  $2^3$ , resultando em 8 combinações de parâmetros (BS, BT, TPS), conforme mostrado na Tabela 1. Cada combinação foi testada em um ambiente controlado, com transações enviadas de um cliente para a rede HLF. Para minimizar o efeito de variáveis externas, os experimentos foram realizados em ordem aleatória. Cada teste teve duração suficiente para garantir a coleta de dados representativos, e a latência média foi registrada para cada configuração.

Os experimentos foram conduzidos em um ambiente de teste configurado com três nós completos do Hyperledger Fabric. A máquina cliente, que submetia as transações, possuía especificações idênticas às dos nós da rede: 4 núcleos físicos, 8 GB de RAM e 100 GB de armazenamento. O ambiente foi configurado utilizando a ferramenta HLF-PET [Moura et al. 2024], e cada nó desempenhou funções de ordenação e validação, conforme a arquitetura proposta por [Mendonça et al. 2023]. A Tabela 1 mostra as combinações de parâmetros para os experimentos e a Figura 3 ilustra o processo de atualização dos parâmetros, assegurando que todas as etapas necessárias sejam seguidas de forma sistemática.

**Tabela 1. Combinações de parâmetros para os experimentos.**

Experimento	BatchSize	BatchTimeout (s)	TPS
1	5	1	5
2	20	1	5
3	5	4	5
4	20	4	5
5	5	1	50
6	20	1	50
7	5	4	50
8	20	4	50

## 5. Resultados

Os resultados dos experimentos foram analisados para identificar os principais fatores que afetam a latência das transações no Hyperledger Fabric. Utilizamos o gráfico de barras para ilustrar a latência média em função das combinações de *BatchSize* (BS), *BatchTimeout* (BT) e *Taxa de Transações por Segundo* (TPS). Complementarmente, aplicamos gráficos de Pareto e efeitos principais para destacar a importância relativa dos fatores, o impacto individual de cada parâmetro e as influências das combinações desses parâmetros no desempenho. Essas análises oferecem uma visão clara e detalhada das variáveis que afetam a eficiência do sistema, proporcionando uma compreensão abrangente de como os parâmetros *BatchSize*, *BatchTimeout* e *TPS* influenciam a latência das transações e o desempenho geral do Hyperledger Fabric.

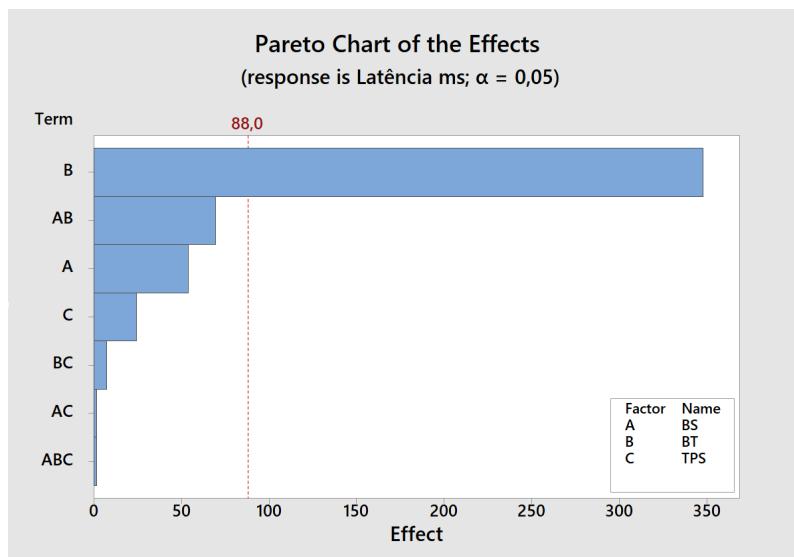
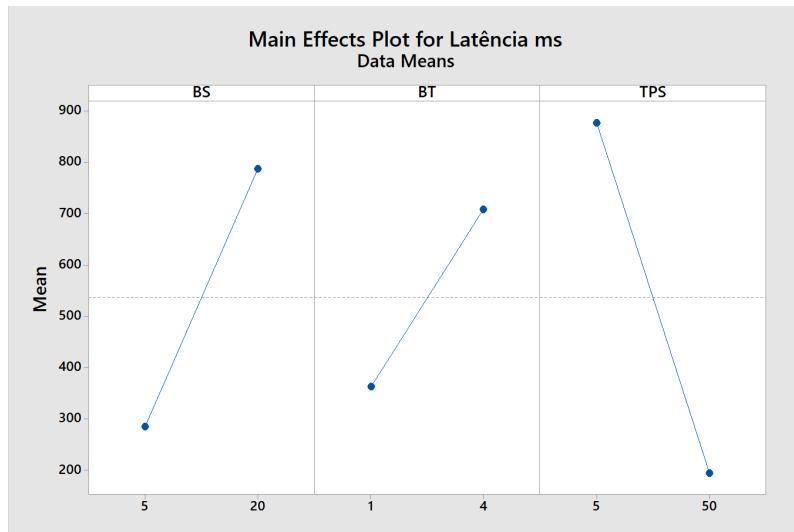


Figura 4. Gráfico de Pareto dos efeitos dos fatores na latência das transações.

O gráfico de Pareto na Figura 4 ilustra a contribuição de cada fator e suas interações para a variação na latência das transações. Os resultados mostram que o fator *BatchTimeout* (BT) tem o maior efeito na latência, com um valor significativamente acima do limite de significância estatística de 88,0. Isso indica que o tempo limite de lote é o fator mais influente na latência das transações no Hyperledger Fabric, sugerindo que ajustar este parâmetro pode resultar em mudanças significativas no desempenho da rede.

A interação entre o tamanho do lote (*BatchSize* - BS) e o tempo limite de lote (*BatchTimeout* - BT), representada como AB, também é substancial, com um efeito notável. Isso sugere que não apenas os valores individuais desses parâmetros são importantes, mas também como eles interagem entre si. O tamanho do lote (*BatchSize* - BS) tem um efeito considerável na latência, seguido pela taxa de transações por segundo (TPS), embora seus impactos sejam menores comparados ao *BatchTimeout*. As interações entre *BatchTimeout* e TPS (BC), *BatchSize* e TPS (AC), e a interação tripla (ABC) têm efeitos menores na latência, todos abaixo do limite de significância.

Os resultados do gráfico de efeitos principais na Figura 5 mostram como os parâmetros *BatchSize* (BS), *BatchTimeout* (BT) e *Taxa de Transações por Segundo* (TPS) impactam a latência média das transações no Hyperledger Fabric. O gráfico revela que a



**Figura 5. Gráfico de Efeitos Principais para a Latência Média das Transações**

latência média aumenta significativamente quando o *BatchSize* (BS) é elevado de 5 para 20, indicando que tamanhos de lote maiores resultam em maior latência devido ao tempo adicional necessário para formar e processar lotes maiores.

Similarmente, um aumento no *BatchTimeout* (BT) de 1 para 4 segundos também resulta em um aumento na latência média, uma vez que tempos limites maiores permitem a acumulação de mais transações antes do processamento, causando atrasos adicionais. Por outro lado, a latência média diminui drasticamente quando a *Taxa de Transações por Segundo* (TPS) aumenta de 5 para 50, sugerindo que taxas de transação mais altas melhoram a eficiência do sistema, provavelmente devido à melhor utilização dos recursos de processamento. Estas observações ressaltam a importância de otimizar esses parâmetros para reduzir a latência e melhorar o desempenho do Hyperledger Fabric. Reduzir o *BatchSize* e o *BatchTimeout* enquanto aumenta a *TPS* pode levar a melhorias significativas na latência das transações.

## 6. Conclusão

Neste artigo, apresentamos uma análise detalhada dos efeitos dos parâmetros *BatchSize* (BS), *BatchTimeout* (BT) e *Taxa de Transações por Segundo* (TPS) no desempenho do Hyperledger Fabric, com foco na latência média das transações. Utilizamos um experimento fatorial  $2^3$  para avaliar sistematicamente como esses parâmetros interagem e influenciam a eficiência da rede. A análise realizada com o gráfico de Pareto revelou que o *BatchTimeout* é o fator com maior impacto na latência, seguido pela interação entre *BatchSize* e *BatchTimeout*. O gráfico de efeitos principais mostrou que a latência média aumenta com valores mais altos de *BatchSize* e *BatchTimeout*, enquanto maiores taxas de transação *TPS* tendem a reduzir essa latência.

Essas descobertas ressaltam a importância de configurar corretamente esses parâmetros para otimizar o desempenho do Hyperledger Fabric, especialmente em aplicações que exigem baixa latência e alta eficiência. Nossa metodologia, baseada em um planejamento experimental fatorial  $2^3$ , demonstrou ser eficaz para quantificar os efeitos dos parâmetros, oferecendo uma base confiável para futuras otimizações. Recomendamos

ajustes específicos no *BatchSize* e *BatchTimeout* como estratégias primárias para minimizar a latência das transações. Além disso, a relação entre os parâmetros deve ser analisada em diferentes cenários de carga para garantir um desempenho consistente da rede.

Para trabalhos futuros, sugerimos explorar métodos dinâmicos de ajuste desses parâmetros, permitindo adaptações automáticas conforme a carga da rede. Além disso, o impacto desses fatores em redes maiores e em cenários mais complexos, como ambientes multi-organizacionais e a integração com contratos inteligentes, deve ser investigado. Também seria relevante analisar como diferentes modelos de consenso podem influenciar a interação desses parâmetros e seus impactos no throughput e na confiabilidade da rede. Outra linha promissora é o uso de modelos preditivos baseados em aprendizado de máquina para aprimorar a otimização de latência e throughput da rede Hyperledger Fabric. Isso permitiria a implementação de sistemas adaptativos, capazes de ajustar os parâmetros em tempo real com base na carga transacional.

Em conclusão, este estudo contribui para uma melhor compreensão dos fatores que afetam a latência das transações no Hyperledger Fabric, fornecendo diretrizes para configuração eficiente e destacando a necessidade de abordagens mais avançadas para aprimorar seu desempenho. Nossa metodologia pode ser replicada por outros pesquisadores e profissionais, impulsionando novas investigações e aprimorando a prática do uso eficiente do Hyperledger Fabric. O desenvolvimento de soluções escaláveis e adaptáveis garantirá que plataformas permissionadas como o Hyperledger Fabric possam atender a requisitos cada vez mais exigentes no cenário das blockchain empresariais.

## Referências

- Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., et al. (2018). Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, page 30. ACM.
- Androulaki, E. and et al. (2018). Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In *Proc. of the EuroSys Conference*.
- Baliga, A., Subhod, I., Kamat, P., and Chatterjee, S. (2018). Performance evaluation of the quorum blockchain platform. *arXiv preprint arXiv:1809.03421*.
- Castro, M., Liskov, B., et al. (1999). Practical byzantine fault tolerance. In *OsDI*, volume 99, pages 173–186.
- Chacko, J. A., Mayer, R., and Jacobsen, H.-A. (2021). Why do my blockchain transactions fail? a study of hyperledger fabric. In *Proceedings of the 2021 international conference on management of data*, pages 221–234.
- de Sá, A. S., Silva Freitas, A. E., and de Araújo Macêdo, R. J. (2013). Adaptive request batching for byzantine replication. *ACM SIGOPS Operating Systems Review*, 47(1):35–42.
- Gaba, P., Raw, R., Mohammed, M. A., Nedoma, J., and Martínek, R. (2022). Impact of block data components on the performance of blockchain-based vanet implemented on hyperledger fabric. *IEEE Access*, 10:71003–71018.

- Greve, F., Sampaio, L., Abijaude, J., Coutinho, A. A., Brito, I., and Queiroz, S. (2018). Blockchain e a Revolução do Consenso sob Demanda. In *Proc. of SBRC Minicursos*.
- Jain, R. (1990). *The art of computer systems performance analysis*. john wiley & sons.
- Kadhm, O. I., Hamad, A. H., and Saeed, F. (2023). High transaction rates performance evaluation for secure e-government based on private blockchain scheme. *Al-Khwarizmi Engineering Journal*.
- Li, D., Jiang, C., Liu, Y., Tang, L., and Yan, L. (2021). A survey on security and performance optimization of blockchain. In *Communications and Networking: 15th EAI International Conference, ChinaCom 2020, Shanghai, China, November 20-21, 2020, Proceedings*, pages 101–111. Springer.
- Liu, C.-M., Badigineni, M., and Lu, S. W. (2021). Adaptive blocksize for iot payload data on fabric blockchain. In *2021 30th Wireless and Optical Communications Conference (WOCC)*, pages 92–96. IEEE.
- Ma, J., Jo, Y., and Park, C. (2020). Peerbft: Making hyperledger fabric's ordering service withstand byzantine faults. *IEEE Access*, 8:217255–217267.
- Melo, C., Oliveira, F., Dantas, J., Araujo, J., Pereira, P., Maciel, R., and Maciel, P. (2022). Performance and availability evaluation of the blockchain platform hyperledger fabric. *The Journal of Supercomputing*, 78(10):12505–12527.
- Mendonça, R., Moura, E., Gonçalves, G., Vieira, A., and Nacif, J. (2023). Comparação e análise de custo e desempenho entre nós de redes blockchain permissionadas e públicas. In *Anais do XLI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 141–154, Porto Alegre, RS, Brasil. SBC.
- Moura, E., Melo, C., Gonçalves, G., Silva, F., and Soares, A. (2024). Uma ferramenta de avaliação de desempenho para plataforma blockchain hyperledger fabric: Hlf-pet. In *Anais do II Colóquio em Blockchain e Web Descentralizada*, pages 8–13, Porto Alegre, RS, Brasil. SBC.
- Roy, U. and Ghosh, N. (2024). Fabman: A framework for ledger storage and size management for hyperledger fabric-based iot applications. *IEEE Transactions on Network and Service Management*.
- Saeed, S. H., Hadi, S., and Hamad, A. H. (2022). Performance evaluation of e-voting based on hyperledger fabric blockchain platform. *Revue d'Intelligence Artificielle*.
- Shalaby, S., Abdellatif, A., Al-Ali, A., Mohamed, A. M., Erbad, A., and Guizani, M. (2020). Performance evaluation of hyperledger fabric. *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, pages 608–613.
- Sousa, J. E. d. A., Oliveira, V., Valadares, J., Dias Goncalves, G., Moraes Villela, S., Soares Bernardino, H., and Borges Vieira, A. (2021). An analysis of the fees and pending time correlation in ethereum. *International Journal of Network Management*.
- Wai, K. and Thein, N. (2023). Performance analysis on block size valuation of hyperledger fabric blockchain. *2023 IEEE Conference on Computer Applications (ICCA)*, pages 50–55.
- Xu, X., Weber, I., and Staples, M. (2019). *Architecture for blockchain applications*. Springer.