

Uma Arquitetura Descentralizada para Aprendizado Federado Baseada em Blockchain: Um Estudo de Caso sobre Mecanismos de Consenso

Francinaldo Barbosa¹, Luis Guilherme Silva¹, Iure Fé¹,
Israel Cardoso¹, Alex Vieira², Geraldo P. Rocha Filho³ e Francisco Airton Silva¹

¹Universidade Federal do Piauí – PI – Brasil

²Universidade Federal de Juiz de Fora – MG – Brasil

³Universidade Estadual do Sudoeste da Bahia – BA – Brasil

{francinaldo.barbosa, luis.e, iure.fe, israel.araujo, faps}@ufpi.edu.br

alex.borges@ufjf.edu.br geraldo.rocha@uesb.edu.br

Resumo. O Aprendizado Federado tem se consolidado como alternativa aos modelos centralizados ao mitigar riscos de privacidade, enquanto sua integração com Blockchain busca eliminar pontos únicos de falha e aumentar a confiança no treinamento distribuído. Contudo, experimentos reais são onerosos e arquiteturas inadequadas podem introduzir ineficiências e vulnerabilidades com impactos significativos em desempenho e segurança. Este trabalho avalia, por meio de simulações no framework FLEX, o impacto dos mecanismos de consenso Proof-of-Work (PoW) e Proof-of-Federated-Learning (PoFL) em sistemas de Aprendizado Federado baseados em Blockchain. Os resultados indicam que o PoFL, orientado ao desempenho do modelo, alcança acurácia superior a 80%, acelera a convergência e reduz o número de rodadas de treinamento, à custa de maior custo computacional por rodada e menor throughput quando comparado ao PoW. Por sua vez, o PoW apresenta validação mais rápida da rede, porém resulta em um processo de aprendizado mais lento e instável. A análise evidencia os trade-offs entre qualidade do modelo, estabilidade do treinamento e custos operacionais.

Abstract. Federated Learning has consolidated itself as an alternative to centralized models by mitigating privacy risks, while its integration with Blockchain seeks to eliminate single points of failure and enhance trust in distributed training. However, real-world experiments are costly, and inadequate architectures may introduce inefficiencies and vulnerabilities with significant impacts on performance and security. This work evaluates, through simulations conducted in the FLEX framework, the impact of the Proof-of-Work (PoW) and Proof-of-Federated-Learning (PoFL) consensus mechanisms on Blockchain-based Federated Learning systems. The results indicate that PoFL, driven by model performance, achieves accuracy above 80%, accelerates convergence, and reduces the number of training rounds, at the expense of higher computational cost per round and lower throughput when compared to PoW. In turn, PoW enables faster network validation but results in a slower and less stable learning process. The analysis highlights the trade-offs among model quality, training stability, and operational costs.

1. Introdução

O Aprendizado Federado (Federated Learning – FL) viabiliza o treinamento colaborativo de modelos de aprendizado de máquina sem a transferência de dados brutos, mantendo as informações nos dispositivos participantes e limitando a comunicação a atualizações de modelo [Xin et al. 2024]. Essa abordagem surge em resposta ao aumento do volume de dados gerados por dispositivos conectados e pela digitalização de serviços, reduzindo a dependência de repositórios centralizados comuns aos métodos tradicionais, nos quais a concentração de dados em servidores remotos amplia a superfície de ataque e cria pontos únicos de falha [Rosano et al. 2022, Yuan et al. 2025]. Apesar desse avanço, a arquitetura predominante do FL mantém a coordenação do treinamento e a agregação global logicamente centralizadas, o que preserva riscos associados à confiança no nó agregador, à disponibilidade do processo e à tolerância a falhas [Li et al. 2021]. Esse cenário tem motivado a investigação da integração entre FL e blockchain, com o objetivo de distribuir a coordenação por meio de mecanismos de consenso que validam coletivamente as atualizações de modelo e as registram em um livro-razão imutável, assegurando integridade e rastreabilidade [Issa et al. 2023, Zhu et al. 2023]. Ainda assim, a adoção dessa integração introduz desafios relacionados à eficiência do treinamento, aos custos computacionais e à escalabilidade de sistemas federados descentralizados, os quais exigem avaliação sistemática.

A integração entre FL e blockchain introduz desafios que afetam diretamente sua aplicação em ambientes reais, em função do aumento da complexidade operacional e dos requisitos de coordenação do sistema [Tian et al. 2023, Zhu et al. 2023]. O uso de mecanismos de consenso distribuídos, combinado ao treinamento contínuo de modelos, eleva os custos computacionais, energéticos e de comunicação, limitando sua adoção em cenários com grande número de participantes ou restrições de recursos [Zhu et al. 2023]. Falhas na concepção ou na implementação dessa integração podem ampliar superfícies de ataque e introduzir vulnerabilidades associadas ao vazamento de informações por meio de atualizações de modelo, metadados ou inconsistências nos protocolos de validação [Issa et al. 2023]. Problemas no desenho do consenso, na governança da rede ou na sincronização entre as camadas de aprendizado e blockchain afetam a confidencialidade, a integridade do treinamento colaborativo e o desempenho do modelo, bem como sua resistência a ataques de envenenamento ou violações de privacidade [Tian et al. 2023]. Apesar do crescente interesse nessa integração, ainda não está claro como diferentes mecanismos de consenso impactam, de forma isolada, a qualidade do aprendizado, a estabilidade do treinamento e os custos operacionais, o que torna necessária uma análise sistemática desses fatores para orientar o uso de consenso em sistemas federados descentralizados [Issa et al. 2023].

A literatura sobre a integração entre FL e Blockchain concentra-se na análise dos mecanismos de consenso como elemento responsável pela coordenação do treinamento, pela validação das atualizações e pela segurança do sistema distribuído [Chen et al. 2021, Liu et al. 2024]. Nesse contexto, o consenso interfere na seleção dos participantes, no aceite das atualizações de modelo e na resistência a comportamentos maliciosos em ambientes federados descentralizados. As propostas incluem consensos voltados à auditoria e à verificação da integridade das atualizações [Wang et al. 2022b], validação de dados com preservação de privacidade [Liu et al. 2025] e estratégias baseadas em esforço computacional, crédito ou histórico de contribuição dos nós [Wang et al. 2022a, Zhao et al. 2024a, Zhao et al. 2024b, Li et al. 2024]. A avaliação desses mecanismos

ocorre majoritariamente por meio de simulações, com uso recorrente de conjuntos de dados de imagem consolidados, como MNIST e suas variações, e métricas associadas ao desempenho do modelo e da rede, incluindo acurácia, latência e *throughput* [Chen et al. 2021, Wang et al. 2022a, Wang et al. 2022b, Li et al. 2024, Liu et al. 2024]. Embora esses estudos demonstrem a viabilidade de diferentes consensos, a diversidade de ambientes experimentais, métricas e arquiteturas dificulta comparações diretas e limita a compreensão isolada do papel do consenso no comportamento de sistemas federados baseados em Blockchain, motivando análises sistemáticas e controladas.

Diante dos desafios impostos pela integração entre FL e Blockchain, este artigo apresenta uma análise comparativa, conduzida por meio de simulação, do impacto de diferentes mecanismos de consenso sobre sistemas federados descentralizados. A abordagem permite investigar de forma sistemática (i) o efeito comparativo dos consensos Proof-of-Work (PoW) e Proof-of-Federated-Learning (PoFL) sobre o desempenho e a dinâmica do FL em um ambiente integrado de aprendizado e Blockchain [Qu et al. 2021], (ii) a avaliação conjunta de métricas relacionadas à qualidade do aprendizado, ao desempenho da rede e à estabilidade do treinamento ao longo das rodadas federadas, e (iii) o impacto da escalabilidade do número de mineradores sobre os custos computacionais e de comunicação associados ao processo de consenso. Os experimentos são conduzidos no *framework* FLEX, que integra FL e Blockchain por meio da biblioteca FLEX-Block, possibilitando a execução de cenários reproduzíveis e alinhados a arquiteturas descentralizadas [Herrera et al. 2025]. Essa escolha contorna a ausência de suporte nativo à coordenação baseada em Blockchain em ferramentas amplamente utilizadas de FL, como FedJAX [Ro et al. 2021] e Flower [Beutel et al. 2020], permitindo isolar o papel do consenso no comportamento do sistema.

O restante deste artigo está organizado da seguinte forma: a Seção 2 discute os conceitos fundamentais que embasam este trabalho; a Seção 3 apresenta uma síntese das principais abordagens da literatura que integram FL e Blockchain; a Seção 4 descreve a arquitetura de FL baseada em Blockchain adotada no *framework*; a Seção 5 apresenta e analisa os resultados obtidos por meio das simulações conduzidas no *framework* FLEX; por fim, a Seção 6 resume os principais achados e discute direções para trabalhos futuros.

2. Background

Esta seção apresenta os fundamentos técnicos e as ferramentas que dão suporte ao desenvolvimento do trabalho.

2.1. Framework FLEX e FLEX-Block

Os experimentos deste trabalho são conduzidos no simulador FLEX, um framework projetado para a prototipação e a avaliação de fluxos de FL em ambientes distribuídos. O FLEX disponibiliza abstrações para a definição de clientes, servidores e estratégias de agregação, organizadas em três módulos funcionais: *data*, *actor* e *pool*. Essa organização permite modelar topologias de rede e comportamentos heterogêneos dos participantes. Permite também lidar com distribuições de dados não independentes e identicamente distribuídas e manter compatibilidade com diferentes bibliotecas de aprendizado de máquina, como PyTorch e TensorFlow, favorecendo a execução de experimentos controlados e reproduzíveis [Herrera et al. 2025].

Integrada ao ecossistema de simulação, a biblioteca FLEX-Block adota uma arquitetura baseada na abstração *BlockchainPool*, responsável por conectar o processo

de treinamento federado ao registro distribuído. Essa camada encapsula as atualizações de pesos em estruturas de blocos genéricas (`_BlockType`). O processo utiliza um esquema de hashing polimórfico compatível com tensores provenientes de diferentes *frameworks*, incluindo PyTorch, TensorFlow e NumPy. A orquestração das trocas de dados ocorre por meio de decoradores especializados, como `@send_weights_to_miner`, que controlam o fluxo de informações e definem as conexões entre clientes e mineradores dentro do `pool`. O ordenamento e a integridade das contribuições no livro-razão são assegurados por protocolos de consenso disponíveis nativamente na biblioteca. Essa organização viabiliza a análise direta da interação entre a coordenação descentralizada e os ciclos de atualização do modelo no FL [Herrera et al. 2025].

2.2. Distribuição Não-IID de Dados em Aprendizado Federado

Um dos principais desafios do FL está associado à heterogeneidade estatística dos dados locais, que raramente satisfazem a suposição de independência e distribuição idêntica entre os participantes. Em ambientes reais, predominam configurações Non-IID, marcadas por desequilíbrios na distribuição de classes, no volume de amostras e nos padrões de coleta, o que faz com que as atualizações locais sigam direções distintas no espaço de parâmetros [Li et al. 2020]. Além disso, essa heterogeneidade costuma coexistir com variações de capacidade computacional, conectividade e disponibilidade dos clientes, ampliando a assimetria no ritmo e na contribuição de cada participante ao longo das rodadas [Kairouz et al. 2021].

Essa divergência entre participantes intensifica o desalinhamento entre as atualizações e caracteriza o fenômeno de *client drift*, no qual o treinamento local tende a se afastar da trajetória esperada para um objetivo global compartilhado. Em termos práticos, clientes com distribuições de dados muito diferentes podem induzir gradientes que “puxam” o modelo para regiões distintas do espaço de parâmetros, gerando oscilações entre rodadas e tornando a convergência menos previsível. Como consequência, a estabilidade do treinamento pode ser impactada, com maior sensibilidade às escolhas de participantes por rodada e ao modo como as contribuições são validadas e incorporadas, especialmente quando a coordenação deixa de depender de um único ponto central [Kairouz et al. 2021].

Para reproduzir esse comportamento de forma controlada, utilizam-se conjuntos de dados Non-IID como o Federated EMNIST [Caldas et al. 2018], que refletem cenários nos quais cada cliente possui um perfil de dados próprio. Essa escolha permite observar, de maneira sistemática, como discrepâncias estatísticas se traduzem em dinâmicas distintas de aprendizado ao longo das rodadas, afetando tanto a consistência das atualizações quanto a evolução do modelo global. No contexto deste trabalho, esse tipo de configuração é particularmente relevante para analisar estabilidade e variação entre rodadas, pois expõe situações em que contribuições locais competem entre si e exigem mecanismos de coordenação capazes de lidar com diferentes tendências de atualização [Kairouz et al. 2021, Li et al. 2020].

3. Trabalhos Relacionados

Nesta seção, são apresentados e discutidos os principais trabalhos que investigam o uso de consensos no FL, destacando a abordagem de avaliação utilizada, qual conjunto de dados foram utilizados, onde e como foram empregados e por fim quais métricas utilizadas para mensuração de desempenho entre os diferentes consensos apresentados em cada tra-

balho. A Tabela 1 resume os trabalhos analisados, agrupando-os conforme os consensos utilizados, avaliação, conjunto de dados, framework e métricas avaliadas.

Tabela 1. Trabalhos Relacionados

Trabalho	Consenso	Tipo de avaliação	Dataset	Framework	Métricas Analisadas
[Wang et al. 2022b]	PoS, PoW, CF-DBFT	Simulação	Dados gerados via simulação	BC-FL	TAC; PSL
[Wang et al. 2022a]	PF-PoFL, PoFL	Protótipo; Simulações	MNIST; CIFAR-10	N/A	Latência de Bloco; Throughput; Computation Cost; Test Accuracy; Contract Running Time
[Liu et al. 2024]	RAFT, FL Raft, PFConsensus, FRConsensus	Simulação	CWRU	N/A	Election Latency; Consensus Latency; System Throughput; Model Accuracy
[Chen et al. 2021]	VBFL-PoS, PoW	Protótipo	MNIST	N/A	Model Accuracy; Malicious Mining Success; Stake Accumulation Rate; Block Generation Time; Latency & Energy; Forking Rate
[Liu et al. 2025]	PoD, CFL, PBFT	Simulação	ImageNet	DBFL	Test Accuracy; Fairness; Complexidade de Comunicação; Resiliência; Safety; Liveliness; Finality
[Li et al. 2024]	PoTW, PoS	Simulação	MNIST; Fashion-MNIST	N/A	Model Accuracy; Distribuição de Stakes; Tempo de Treinamento
[Zhao et al. 2024a]	PoA, PoSap, PF-PoFL, PoS, DPoS, CPoS	Simulação	EUA; MNIST	BFL	Global Contribution; Efficiency; Acurácia do modelo local; Data-related delay; Model-related delay; Aggregation delay; Consensus delay
[Zhao et al. 2024b]	PoA, PoSap, PF-PoFL, LPoS	Simulação	EUA; MNIST	BFL	Long-term Contribution; Long-term Efficiency; Long-term Accuracy
Este Trabalho	PoS, PoFL	Simulação	Federated EM-NIST	FLEX	Loss; Acurácia; Throughput; Drift de Pesos; Variação de mineradores

Mecanismo de Consenso. Os trabalhos analisados empregam diferentes estratégias para validar a legitimidade das atualizações e selecionar os nós responsáveis pela manutenção da rede, propondo, em sua maioria, mecanismos próprios ou variações estruturais dos consensos existentes. A validação direta da qualidade ou integridade do modelo é uma abordagem central em estudos como [Chen et al. 2021] e [Liu et al. 2024], que utilizam mecanismos de stake (participação) e avaliação de modelos para mitigar ataques de envenenamento. Em uma linha voltada à integridade e auditoria, [Wang et al. 2022b] propõe um sistema de auditoria por correspondência, enquanto [Liu et al. 2025] foca na verificação de dados com preservação de privacidade. Já a avaliação baseada no esforço e na contribuição histórica do participante é explorada por [Zhao et al. 2024a] e [Zhao et al. 2024b], que introduzem métricas de contribuição de longo prazo e consciência de contexto para garantir justiça no sistema. Por fim, as abordagens de [Wang et al. 2022a] e [Li et al. 2024] utilizam sistemas de crédito e Prova de Trabalho de Treinamento (PoTW) para incentivar o desempenho dos nós. Este trabalho diferencia-se ao realizar um estudo de caso focado nos consensos PoW e PoFL, ambos consolidados na literatura, utilizando um framework que integra FL e Blockchain

para analisar o comportamento e a viabilidade dessas abordagens frente aos desafios de avaliação e consenso.

Tipo de Avaliação. A validação das propostas apresentadas na literatura concentra-se majoritariamente em ambientes de simulação, embora existam abordagens que avançam para a implementação de protótipos. Trabalhos como [Wang et al. 2022b], [Liu et al. 2024], [Li et al. 2024], [Zhao et al. 2024a], [Zhao et al. 2024b] e [Liu et al. 2025] utilizam estritamente a simulação para avaliar o desempenho de seus algoritmos de consenso e a segurança da rede. Por outro lado, as pesquisas de [Wang et al. 2022a] e [Chen et al. 2021] combinam simulações com o desenvolvimento de protótipos, permitindo uma análise mais próxima de cenários reais de execução. Independentemente do método, o foco comum dessas avaliações é verificar a viabilidade técnica dos consensos propostos em redes descentralizadas. Este trabalho segue a tendência predominante da literatura ao utilizar a simulação como método de avaliação, porém diferencia-se ao aplicar esse método em um estudo de caso comparativo rigoroso entre PoW e PoFL, utilizando um framework customizado para analisar métricas de desempenho e comportamento dos mineradores.

Conjunto de Dados. A avaliação da eficácia dos modelos de aprendizado e dos mecanismos de consenso baseia-se, em sua maioria, em conjuntos de dados de imagem amplamente reconhecidos pela comunidade científica. O dataset MNIST é o mais frequente, sendo utilizado de forma isolada ou combinada em trabalhos como [Chen et al. 2021], [Wang et al. 2022a], [Li et al. 2024], [Zhao et al. 2024a] e [Zhao et al. 2024b]. Além deste, o CIFAR-10 é empregado por [Wang et al. 2022a] para testes de maior complexidade, enquanto variações como o Fashion-MNIST são exploradas por [Li et al. 2024]. Outros estudos utilizam dados específicos de domínios distintos, como o ImageNet para reconhecimento visual em larga escala [Liu et al. 2025], o dataset CWRU para diagnóstico de falhas [Liu et al. 2024], ou dados gerados via simulação e de sensores urbanos (EUA) [Wang et al. 2022b, Zhao et al. 2024a]. Este trabalho diferencia-se ao utilizar o Federated EMNIST, um conjunto de dados que, embora baseado nos caracteres do MNIST, é estruturado especificamente para refletir a distribuição de dados não-independentes e identicamente distribuídos (Non-IID), típica de cenários reais de FL.

Framework. A estruturação dos ambientes de teste revela uma divisão entre o uso de arquiteturas customizadas e frameworks específicos para a integração de Blockchain e FL. Diversos estudos, como [Chen et al. 2021, Wang et al. 2022a, Li et al. 2024, Liu et al. 2024], optam por implementações próprias baseadas em bibliotecas padrão de mercado, como PyTorch e outras bibliotecas de suporte, sem a adoção de um framework nomeado. Em contrapartida, outras pesquisas propõem ou utilizam estruturas voltadas especificamente à descentralização e orquestração, como o BC-FL [Wang et al. 2022b], o DBFL [Liu et al. 2025] e o framework BFL, adotado por [Zhao et al. 2024a] e [Zhao et al. 2024b]. Este trabalho destaca-se ao utilizar o framework FLEX, uma ferramenta robusta que permite a criação de simulações precisas de fluxos federados, proporcionando o suporte necessário com sua biblioteca FLEX-Block para a execução do estudo de caso entre os consensos PoW e PoFL em um ambiente controlado, replicável e especializado para experimentos de FL.

Métricas Analisadas. A avaliação dos sistemas propostos utiliza um conjunto diversificado de indicadores que buscam equilibrar a eficiência da rede com a qualidade do aprendizado. A Acurácia do Modelo (Model Accuracy) é a métrica onipresente,

aparecendo em quase todos os trabalhos, como [Chen et al. 2021, Wang et al. 2022a, Wang et al. 2022b, Zhao et al. 2024a, Li et al. 2024, Liu et al. 2024], muitas vezes acompanhada pela análise de latência (consenso, eleição e bloco) e throughput do sistema para medir a escalabilidade da blockchain. Pesquisas focadas em governança e justiça, como as de [Zhao et al. 2024b] e [Liu et al. 2025], introduzem métricas de contribuição de longo prazo e justiça (fairness). Este trabalho diferencia-se por apresentar um conjunto holístico de métricas que cobrem tanto o desempenho do modelo, através da Acurácia e Loss, quanto a eficiência da rede via Throughput. Além disso, introduz a análise de Drift de Pesos e a Variação de mineradores, permitindo uma compreensão mais profunda de como a alternância de consensos (PoW e PoFL) afeta a estabilidade e a convergência do treinamento federado.

4. Arquitetura

A arquitetura proposta organiza o sistema em três camadas funcionais: agregação federada local, validação distribuída e consenso, estruturando o processo de treinamento de forma modular e bem delimitada. Essa organização permite analisar de maneira isolada o efeito do mecanismo de consenso sobre o treinamento federado, sem interferir diretamente nas demais etapas do fluxo de aprendizado. Ao separar explicitamente essas responsabilidades, a arquitetura facilita a compreensão das interações entre aprendizado, coordenação e validação em ambientes descentralizados.

A separação em camadas dissocia o aprendizado local executado pelos clientes, a troca de informações entre os validadores na rede distribuída e a decisão global de aceitação do modelo no nível do consenso. Com isso, torna-se possível comparar diferentes protocolos de consenso e observar suas implicações no comportamento do sistema, na estabilidade do treinamento e na dinâmica de convergência, mantendo inalterado o fluxo federado subjacente. Essa abordagem favorece análises controladas e contribui para identificar, de forma mais clara, o papel específico do consenso na evolução do modelo global.

A arquitetura é instanciada em um ambiente de simulação baseado no framework FLEX e na biblioteca FLEX-Block¹, que integra FL descentralizado e blockchain em uma topologia híbrida. A Figura 1 apresenta a organização conceitual do sistema. Em lugar de um servidor central, a estrutura adota um conjunto de mineradores logicamente equivalentes, responsáveis pela agregação federada regional e pela validação distribuída. Essa descentralização remove o ponto único de falha e permite a execução de protocolos de consenso sobre modelos treinados de forma federada, sem dependência de uma entidade central de controle.

A primeira camada corresponde à agregação federada local, responsável pela execução do ciclo de aprendizado nos dispositivos participantes. Nessa camada, os clientes realizam treinamento local a partir de um modelo global compartilhado, utilizando partições de dados heterogêneas. As atualizações produzidas refletem tanto as características locais dos dados quanto a dinâmica de treinamento de cada cliente. Essas contribuições são encaminhadas ao minerador responsável, que atua como agregador regional e consolida as atualizações em um modelo candidato representativo daquela região. O processo ocorre de forma paralela entre os mineradores, preservando o isolamento en-

¹<https://github.com/FLEXible-FL/FLEXible>
<https://github.com/FLEXible-FL/flex-block>

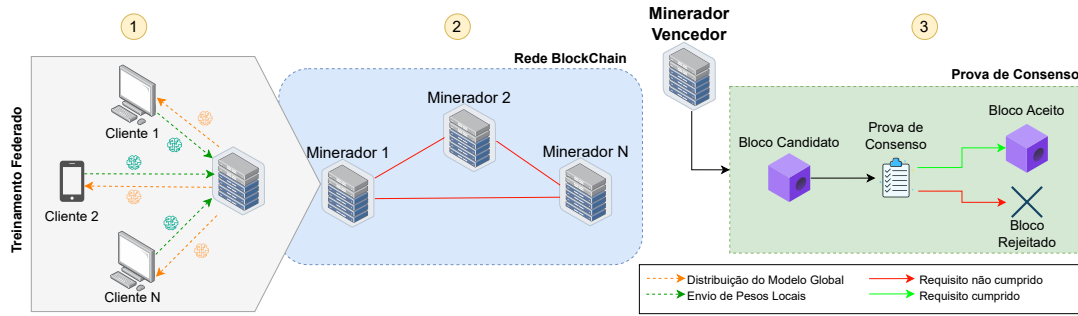


Figura 1. Arquitetura conceitual do fluxo de simulação e validação

tre os conjuntos de clientes e garantindo que apenas parâmetros agregados avancem para as etapas de validação e consenso.

A segunda camada define o processo de validação distribuída e a comunicação global entre os mineradores, sustentado por uma rede *peer-to-peer*. Nessa etapa, os modelos candidatos gerados localmente deixam de ser entidades regionais e passam a integrar um domínio de visibilidade global. Cada minerador mantém uma visão consistente do estado do sistema e participa da troca de informações necessárias para a validação das propostas submetidas. Esse processo garante que todos os participantes tenham acesso aos mesmos candidatos antes da decisão de consenso, assegurando sincronização, consistência e ordenação das propostas ao longo das rodadas. A camada de validação distribuída estabelece, assim, a transição entre a agregação local e a seleção global do modelo.

A terceira camada compreende os mecanismos de governança e consenso responsáveis por selecionar, entre os modelos candidatos validados, aquele que será incorporado ao estado global do sistema. O consenso define as regras segundo as quais os mineradores competem ou cooperam para a aceitação das propostas, determinando o critério de seleção do bloco vencedor e o encadeamento das rodadas de treinamento. Ao registrar o modelo aceito em um livro-razão distribuído, essa camada assegura integridade, ordenação temporal e imutabilidade do estado global. Diferentes estratégias de consenso podem ser adotadas nessa camada, alterando o critério de decisão sem modificar o fluxo de FL ou a estrutura de comunicação estabelecida nas camadas anteriores.

Do ponto de vista arquitetural, a principal diferença entre os consensos PoW e PoFL reside exclusivamente no critério de seleção do bloco vencedor. Em ambos os casos, mantêm-se inalterados o fluxo de agregação federada local, a propagação *peer-to-peer* entre os mineradores e a estrutura de validação distribuída. Essa característica permite isolar o mecanismo de consenso como a principal variável experimental, facilitando a análise comparativa apresentada na Seção 5. Para formalizar esse comportamento e explicitar as dependências entre aprendizado local, validação distribuída e consenso, a subseção 4.1 apresenta a formulação da arquitetura, modelando o processo federado em rodadas discretas e caracterizando matematicamente o papel do consenso na evolução do estado global do sistema.

4.1. Formulação da Arquitetura

Na arquitetura, o processo de treinamento federado descentralizado é modelado em rodadas discretas $t = 0, \dots, T - 1$, sobre um conjunto de mineradores \mathcal{M} e clientes \mathcal{C} , particionados de forma disjunta em subconjuntos \mathcal{C}_m associados a cada minerador $m \in \mathcal{M}$.

Em cada rodada, os clientes recebem o modelo global corrente w_t e realizam treinamento local sobre seus conjuntos de dados D_i , produzindo atualizações $w_{t+1}^{(i)}$. Cada minerador atua como agregador regional e sintetiza um modelo candidato $\tilde{w}_{t+1}^{(m)}$ por meio do algoritmo FedAvg, calculado como uma média ponderada das atualizações dos clientes sob sua coordenação, conforme a Equação (1).

$$\tilde{w}_{t+1}^{(m)} = \sum_{i \in \mathcal{C}_m} \frac{|D_i|}{\sum_{j \in \mathcal{C}_m} |D_j|} w_{t+1}^{(i)} \quad (1)$$

Os modelos candidatos $\tilde{w}_{t+1}^{(m)}$ são então propagados na rede *peer-to-peer* e submetidos ao mecanismo de consenso da blockchain. No consenso PoFL, a seleção do bloco vencedor é formulada como a maximização de uma função de desempenho $S(\cdot)$, avaliada sobre um conjunto de validação global D_{eval} , cujo custo cresce com $|D_{\text{eval}}|$ e pode tornar-se um gargalo em larga escala, mitigável por amostragem ou delegação da avaliação, de modo que o minerador vencedor m_t^* satisfaz a Equação (2). Em contraste, no consenso PoW, o minerador vencedor é definido pela resolução de um desafio criptográfico, independentemente da qualidade do modelo agregado, de forma que a seleção de m_t^* é dissociada do desempenho do aprendizado.

$$m_t^* = \arg \max_{m \in \mathcal{M}} S\left(\tilde{w}_{t+1}^{(m)}\right) \quad (2)$$

Uma vez definido o minerador vencedor, o modelo correspondente é atualizado como estado global do sistema na rodada seguinte, conforme a Equação (3), sendo encapsulado em um novo bloco e adicionado à blockchain. Todos os mineradores sincronizam seu estado interno com o modelo validado, garantindo consistência global antes do início da próxima rodada de treinamento. Essa formulação evidencia que, mantendo-se fixos o processo de agregação federada e a comunicação entre os participantes, o mecanismo de consenso constitui a principal variável que governa a dinâmica de convergência, estabilidade e custo operacional da arquitetura.

$$w_{t+1} = \tilde{w}_{t+1}^{(m_t^*)} \quad (3)$$

4.2. Métricas de Estabilidade e Desempenho

Para avaliar o impacto dos protocolos de consenso na dinâmica do sistema, a arquitetura simulada incorpora métricas quantitativas que monitoram a convergência e a eficiência operacional:

1. **Weight Drift (Δ_w):** Mensura a estabilidade do aprendizado entre rodadas sucessivas. É calculada pela norma Euclidiana (L_2) entre os pesos do modelo no bloco atual w_t e os pesos do bloco imediatamente anterior w_{t-1} :

$$\Delta_w = \|w_t - w_{t-1}\|_2$$

Esta métrica permite identificar se a seleção agnóstica do PoW introduz divergências estocásticas no espaço de parâmetros que retardam a convergência global.

2. **Throughput de Blocos (Bps):** Define a vazão do sistema em blocos por segundo. É calculado pela razão entre o número total de blocos minerados $|\mathcal{B}|$ e o tempo total acumulado de simulação $\sum \tau$:

$$Bps = \frac{|\mathcal{B}|}{\sum_{t=1}^T \tau_{\text{exec}}(t)}$$

A análise desta métrica evidencia o custo temporal imposto pela camada de blockchain, contrastando o esforço computacional do PoW com o tempo de inferência exigido pelo PoFL.

5. Estudo de Casos

Esta Seção apresenta a análise dos resultados obtidos a partir de simulações realizadas no ambiente FLEX, com o propósito de comparar o comportamento dos mecanismos de consenso PoW e PoFL. A análise concentra-se no efeito desses consensos sobre três dimensões da arquitetura descrita na Seção 4: eficiência do aprendizado, escalabilidade da rede e estabilidade do modelo. Os experimentos utilizam o conjunto de dados *Federated EMNIST* e o algoritmo de agregação FedAvg, mantendo fixa a arquitetura do modelo, baseada em uma rede neural convolucional simples. O cenário inicial considera um total de 3579 clientes federados, distribuídos entre 10 mineradores, sendo nove responsáveis por 358 clientes e um por 357, com treinamento realizado em uma única época. As métricas de *loss*, acurácia, *throughput* e *drift* de pesos são avaliadas ao longo de 10 rodadas de treinamento, de modo que a única variável controlada seja o mecanismo de consenso. Em seguida, a análise de escalabilidade é conduzida por meio da variação do número de mineradores na rede.

A Figura 2 compara o comportamento dos consensos PoFL e PoW a partir da evolução conjunta da acurácia e da função de perda ao longo do treinamento. A Figura 2a apresenta a trajetória da acurácia de teste global ao longo de 10 rodadas, correspondente à acurácia do modelo global validado na blockchain e avaliado em um conjunto de teste global fixo, externo ao treinamento federado. Observa-se que o PoFL alcança valores acima de 80% nas rodadas iniciais, mantendo uma progressão estável ao longo do processo. Em contraste, o PoW apresenta crescimento gradual, acompanhado de flutuações entre rodadas, e encerra o treinamento com acurácia inferior. Essa diferença reflete diretamente o critério de seleção do consenso. Ao priorizar modelos com melhor desempenho técnico durante a validação, o PoFL favorece atualizações mais alinhadas à convergência do modelo global, enquanto o PoW permanece condicionado a fatores independentes da qualidade do aprendizado.

A Figura 2b apresenta o comportamento da Perda Média (*Loss*), utilizando a função *CrossEntropyLoss*, e confirma o desempenho observado na métrica de acurácia. Nota-se que o mecanismo PoFL consegue reduzir o erro do modelo de forma muito mais agressiva e estável, aproximando-se de valores abaixo de 1.0, enquanto o PoW mantém uma trajetória de queda linear mais lenta e com valores de perda consideravelmente maiores ao final das mesmas 10 rodadas. A análise conjunta de ambas as figuras torna explícita a diferença que as provas de consenso trazem para a arquitetura: ao vincular a validação do bloco à eficácia do aprendizado, o PoFL otimiza o ciclo de vida do modelo de maneira que o PoW, focado em esforço computacional genérico, não consegue replicar.

A Figura 3 analisa o desempenho da rede sob diferentes condições operacionais,

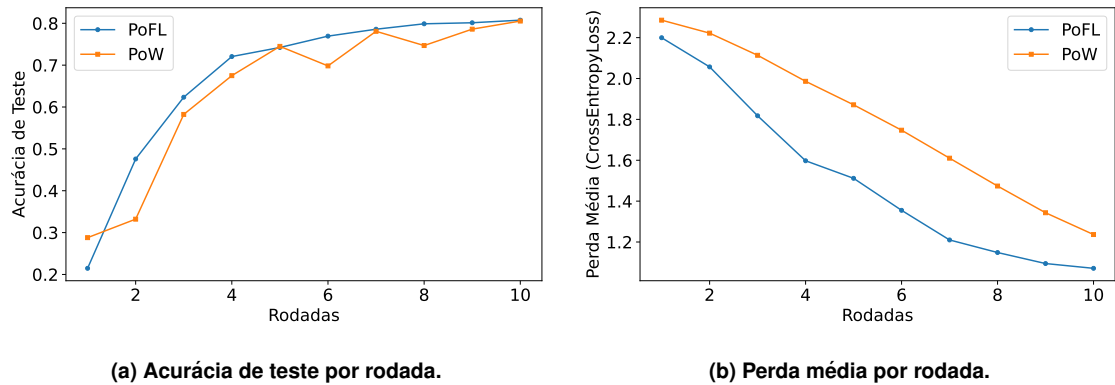


Figura 2. Comparativo de convergência do modelo global.

considerando tanto a evolução do processamento ao longo das rodadas quanto o impacto da variação no número de mineradores, de modo a caracterizar a eficiência e a escalabilidade dos mecanismos de consenso. A Figura 3a detalha o *throughput* da rede, medido em blocos por segundo (B/s), ao longo das rodadas de treinamento em um cenário com número fixo de mineradores. Observa-se que o PoW mantém uma taxa de processamento significativamente superior e mais estável ao longo do tempo, estabilizando-se em torno de 0.25 B/s. Em contraste, o PoFL apresenta um *throughput* consideravelmente menor, operando em patamares próximos a 0.03 B/s. Esse comportamento decorre do custo adicional imposto pelo mecanismo PoFL, que incorpora, ao processo de consenso, a execução do treinamento agregado e a avaliação de desempenho do modelo antes do fechamento de cada bloco. No contexto da simulação adotada no FLEX, esse custo passa a dominar o tempo total da rodada, reduzindo o *throughput* da rede. Em contraste, no PoW, o tempo associado à validação criptográfica do bloco permanece limitado e não depende do número de modelos candidatos ou de avaliações adicionais, o que resulta em uma taxa de processamento mais elevada ao longo das rodadas.

Em sequência, a Figura 3b analisa o tempo médio por rodada em função da quantidade de mineradores, mantendo fixos os demais parâmetros da simulação, como métrica de escalabilidade da arquitetura. Observa-se que, no contexto da simulação adotada, o PoW mantém um tempo de execução baixo e praticamente constante, em torno de 5 segundos por rodada, mesmo com o aumento do número de mineradores. Esse comportamento decorre do fato de que, no modelo implementado pelo FLEX, o custo temporal do consenso PoW não é diretamente influenciado pela quantidade de participantes, uma vez que aspectos como latência de propagação e competição entre blocos não constituem o fator dominante do tempo por rodada. Em contraste, o PoFL apresenta crescimento linear do tempo por rodada, que aumenta de aproximadamente 28 segundos com 10 mineradores para mais de 80 segundos com 35 mineradores, em razão do aumento no número de modelos candidatos avaliados durante a etapa de consenso. Esse comportamento evidencia que, embora o PoFL favoreça modelos com melhor qualidade de aprendizado, tal estratégia impõe um custo temporal que cresce com o número de participantes da validação.

A Figura 4 apresenta o comportamento do Weight Drift, métrica que quantifica a variação absoluta dos pesos do modelo entre blocos consecutivos, permitindo avaliar a estabilidade das atualizações impostas pelos mecanismos de consenso. No PoFL, observa-

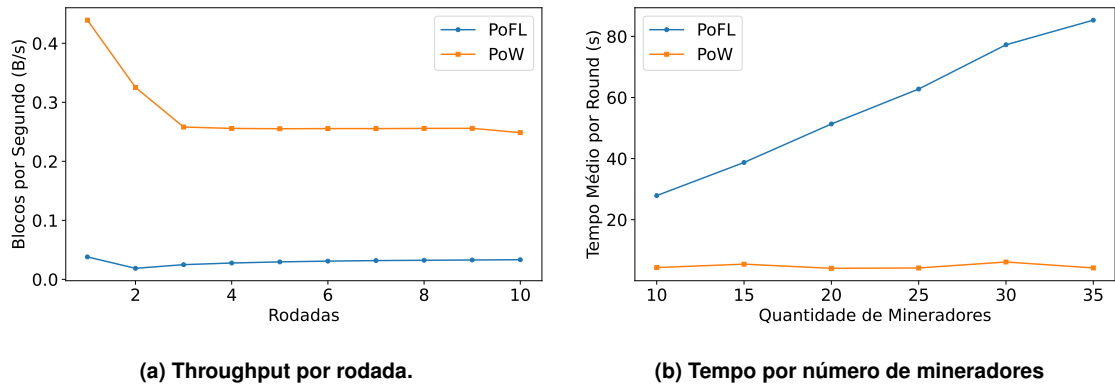


Figura 3. Análise de desempenho e escalabilidade da rede.

se uma deriva praticamente nula nas três rodadas iniciais, caracterizando uma fase de convergência inicial com atualizações mínimas. Entre as rodadas 4 e 7, surgem picos pronunciados e oscilações mais intensas, superiores às observadas no PoW. Esse comportamento está associado à natureza competitiva do PoFL, na qual diferentes mineradores, treinados sobre subconjuntos Non-IID de dados, alternam-se como vencedores do consenso com base no desempenho de seus modelos agregados, promovendo substituições mais abruptas do modelo global.

A partir da sétima rodada, o PoFL passa a apresentar redução progressiva da deriva de pesos, acompanhada por um ritmo mais acentuado de aproximação entre modelos sucessivos. Essa tendência se mantém até o término do treinamento, quando a deriva se aproxima de zero, indicando maior alinhamento entre as atualizações agregadas. Em contraste, o PoW mantém valores de deriva relativamente constantes ao longo das rodadas, resultado de um processo de seleção do minerador baseado em critérios criptográficos, dissociados do desempenho do modelo. Esse comportamento leva a atualizações com menor variação entre rodadas. O conjunto dos resultados aponta que, apesar de introduzir maior dispersão na fase intermediária, o PoFL conduz o treinamento a um estado final com maior coerência entre os pesos agregados.

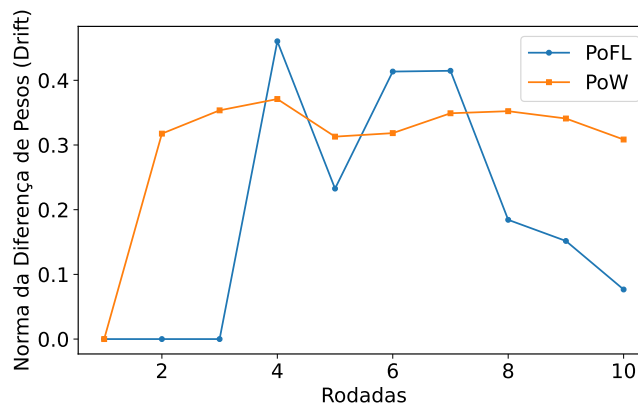


Figura 4. Weight drift por rodada.

6. Conclusão

O estudo de casos avaliou, por meio de simulações no *framework* FLEX, como dois diferentes mecanismos de consenso afetam o funcionamento de sistemas de FL apoiados por Blockchain, acompanhando a evolução de métricas de aprendizado, desempenho da rede e estabilidade do modelo ao longo do treinamento. Os resultados indicam que o tipo de consenso interfere diretamente na trajetória de convergência: abordagens orientadas ao desempenho do aprendizado, como o PoFL, alcançam níveis mais altos de acurácia em menos rodadas e apresentam redução de perda mais regular, enquanto mecanismos baseados em esforço computacional, como o PoW, exibem evolução mais gradual e com menor variação entre rodadas. Os custos associados à validação também se mostraram relevantes, pois o PoFL impõe maior tempo médio por rodada e menor *throughput*, efeito que se acentua à medida que o número de mineradores cresce, ao passo que o PoW mantém comportamento mais previsível sob expansão da rede. A análise da deriva de pesos reforça essa distinção, ao indicar maior variação intermediária nos consensos dependentes da qualidade do modelo, seguida por um estado final com maior alinhamento entre atualizações sucessivas. Como limitação do estudo, destaca-se o fato de o FLEX ser um *framework* recente, com um conjunto ainda restrito de mecanismos de consenso disponíveis de forma nativa, o que limita a abrangência das comparações realizadas. Ainda assim, o ambiente de simulação permite análises controladas e reprodutíveis, oferecendo uma base consistente para a extensão futura do estudo com novos consensos e métricas voltadas à segurança e à resiliência do processo federado.

Referências

- Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., Sani, L., Li, K. H., Parcollet, T., De Gusmão, P. P. B., et al. (2020). Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390*.
- Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konečný, J., McMahan, H. B., Smith, V., and Talwalkar, A. (2018). Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*.
- Chen, H., Asif, S. A., Park, J., Shen, C., and Bennis, M. (2021). Robust blockchained federated learning with model validation and proof-of-stake inspired consensus. *CoRR*, abs/2101.03300.
- Herrera, F., Jiménez-López, D., Argente-Garrido, A., Rodríguez-Barroso, N., Zuheros, C., Aguilera-Martos, I., Bello, B., García-Márquez, M., and Luzón, M. V. (2025). Flex: Flexible federated learning framework. *Information Fusion*, 117:102792.
- Issa, W., Moustafa, N., Turnbull, B., Sohrabi, N., and Tari, Z. (2023). Blockchain-based federated learning for securing internet of things: A comprehensive survey. *ACM Computing Surveys*, 55(9):1–43.
- Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2021). Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210.
- Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., Liu, X., and He, B. (2021). A survey on federated learning systems: Vision, hype and reality for data privacy and protection. *IEEE Transactions on Knowledge and Data Engineering*, 35(4):3347–3366.

- Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60.
- Li, Y., Xia, C., Lin, W., and Wang, T. (2024). Ppbfl: A privacy protected blockchain-based federated learning model. *arXiv preprint arXiv:2401.01204*.
- Liu, H., Zhu, F., and Cheng, L. (2025). Proof-of-data: A consensus protocol for collaborative intelligence. *arXiv preprint arXiv:2501.02971*.
- Liu, L., Hu, Y., Zhao, Y., Zhang, X., Ma, Y., and Chang, G. (2024). A novel federated learning system with privacy protection and blockchain consensus incentive mechanisms in cloud-edge collaboration scenarios. In *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3111–3118.
- Qu, X., Wang, S., Hu, Q., and Cheng, X. (2021). Proof of federated learning: A novel energy-recycling consensus algorithm. *IEEE Transactions on Parallel and Distributed Systems*, 32(8):2074–2085.
- Ro, J. H., Suresh, A. T., and Wu, K. (2021). Fedjax: Federated learning simulation with jax. *arXiv preprint arXiv:2108.02117*.
- Rosano, B., Bittencourt, L., Anjos, J., Técnico-IC-PFG, R., and de Graduação, P. F. (2022). Aprendizado federado hierárquico.
- Tian, Y., Guo, Z., Zhang, J., and Al-Ars, Z. (2023). Dfl: High-performance blockchain-based federated learning. *Distributed Ledger Technologies: Research and Practice*, 2(3):1–25.
- Wang, Y., Peng, H., Su, Z., Luan, T. H., Benslimane, A., and Wu, Y. (2022a). A platform-free proof of federated learning consensus mechanism for sustainable blockchains. *IEEE Journal on Selected Areas in Communications*, 40(12):3305–3324.
- Wang, Y., Zhou, J., Feng, G., Niu, X., and Qin, S. (2022b). Blockchain assisted federated learning for enabling network edge intelligence. *IEEE Network*, 37(1):96–102.
- Xin, W., Jiaqian, L., Xueshuang, D., Haoji, Z., and Lianshan, S. (2024). A survey of differential privacy techniques for federated learning. *IEEE Access*.
- Yuan, J., Liu, W., Shi, J., and Li, Q. (2025). Approximate homomorphic encryption based privacy-preserving machine learning: a survey. *Artificial Intelligence Review*, 58(3):82.
- Zhao, Y., Qu, Y., Xiang, Y., Chen, F., and Gao, L. (2024a). Context-aware consensus algorithm for blockchain-empowered federated learning. *IEEE Transactions on Cloud Computing*, 12(2):491–503.
- Zhao, Y., Qu, Y., Xiang, Y., Chen, F., and Gao, L. (2024b). Long-term proof-of-contribution: An incentivized consensus algorithm for blockchain-enabled federated learning. *IEEE Transactions on Services Computing*, 17(5):2558–2570.
- Zhu, J., Cao, J., Saxena, D., Jiang, S., and Ferradi, H. (2023). Blockchain-empowered federated learning: Challenges, solutions, and future directions. *ACM Computing Surveys*, 55(11):1–31.