Divergência de Peso em Aprendizado Federado: Impacto no Modelo Global Considerando Dados Non-IID

Deivision Oliveira
Instituto de Informática
UFG
Goiânia, Brasil
deivison@discente.ufg.br

Mateus Souza
Instituto de Informática
UFG
Goiânia, Brasil
moovcont@gmail.com

Ana Luísa de Bastos Chagas
Instituto de Informática
UFG
Goiânia, Brasil
analuisa23@discente.ufg.br

Maurício Rodrigues Lima
Instituto de Informática
UFG
Goiânia, Brasil
k20x@outlook.com

Antonio Oliveira
Instituto de Informática
UFG
Goiânia, Brasil
antoniojr@ufg.br

Renan Rodrigues de Oliveira
Instituto Federal de Goiás
Goiânia, Brasil
renan.rodrigues@ifg.edu.br

Deller Ferreira
Instituto de Informática
UFG
Goiânia, Brasil
deller@inf.ufg.br

Elisângela Silva Dias
Instituto de Informática
UFG
Goiânia, Brasil
elisangelasd@ufg.br

***Abstract*—This paper investigates how weight divergence can serve as a reliable indicator of a device's contribution to the global model in Federated Learning (FL) systems, enabling the omission of model transmission when the contribution is minimal. This study is divided into two main parts. First, an analysis of the device scheduling method presented in the article "Federated Learning Over Wireless IoT Networks" was conducted. Second, we propose an algorithm that employs L2 regularization for scheduling ultra-constrained IoT devices, considering data heterogeneity (non-IDD data), with the goal of reducing energy consumption by minimizing the number of transmissions of outdated models to the central server. The results indicate that the proposed approach can optimize energy efficiency and communication in IoT networks compared to traditional methods.**

***Keywords*—Federated Learning; Non-IDD; Weight Divergence.**

***Resumo*—Este estudo investiga como a divergência de peso pode servir como um indicador confiável da contribuição de um dispositivo para o modelo global em sistemas de Aprendizado Federado (AF), permitindo a omissão da transmissão do modelo quando a contribuição é mínima. Este estudo é dividido em duas partes principais. Primeiro, foi realizada uma análise do método de agendamento de dispositivos apresentado no artigo "Federated Learning Over Wireless IoT Networks". Em seguida, este estudo propõe um algoritmo que emprega regularização L2 para o agendamento de dispositivos IoT ultrarrestritos, considerando a heterogeneidade dos dados (dados *no-IDD*), com o objetivo de reduzir o consumo de energia minimizando o número de transmissões de modelos desatualizados para o servidor central. Os resultados indicam que a abordagem proposta pode otimizar a eficiência energética e a comunicação em redes IoT em comparação com os métodos tradicionais.**

***Palavras-chave*—Aprendizado Federado; Non-IDD; Divergência de Peso.**

I. Introdução

O Aprendizado Federado (AF) é uma abordagem descentralizada para o treinamento de modelos de aprendizado de máquina, sendo especialmente vantajosa em cenários que envolvem dispositivos distribuídos, como *smartphones* e sensores de Internet das Coisas (*Internet of Things* – IoT) [1]. Diferentemente do treinamento centralizado tradicional, que requer a agregação de todos os dados em um único local, o AF permite que as atualizações do modelo sejam realizadas localmente nos dispositivos, compartilhando apenas os gradientes ou atualizações resultantes. Essa metodologia aprimora a privacidade dos dados, reduz riscos de violações de segurança e minimiza a necessidade de transferência de informações sensíveis para servidores centrais [2].

Apesar de suas vantagens, o AF apresenta desafios significativos, principalmente devido à heterogeneidade dos dispositivos participantes e dos dados coletados [3]. Diferenças na capacidade computacional, na conectividade e na distribuição dos dados podem resultar em contribuições desiguais para o modelo global, impactando sua convergência e desempenho geral. Um dos principais desafios nesse contexto é avaliar a relevância das atualizações locais, garantindo que apenas as mais significativas sejam incorporadas ao modelo global. Uma

métrica promissora para enfrentar esse problema é a análise da divergência de peso ao longo das rodadas de treinamento, pois fornece compreensão sobre a eficácia e o impacto de cada atualização local [4].

Neste estudo, investigamos o uso da divergência de peso como critério para otimizar a eficiência da comunicação no AF. Ao identificar e priorizar atualizações significativas do modelo, buscamos reduzir a transmissão de modelos locais redundantes ou obsoletos, diminuindo o consumo de recursos de rede e melhorando a eficiência geral do sistema. Para isso, propomos um algoritmo que incorpora regularização L2 para o agendamento de atualizações em dispositivos IoT ultrarrestritos [5]. Essa abordagem visa minimizar a transmissão de modelos desatualizados, garantindo que apenas atualizações relevantes sejam propagadas no processo de aprendizado global, o que contribui para um melhor desempenho do sistema como um todo.

A metodologia proposta é especialmente relevante para aplicações reais de AF, onde a otimização do uso da rede e da energia é essencial para garantir escalabilidade e sustentabilidade. Ao selecionar de forma criteriosa quais modelos locais devem ser enviados em cada rodada de treinamento, nossa abordagem não apenas reduz a sobrecarga na comunicação, mas também prolonga a vida útil de dispositivos IoT alimentados por bateria. O código-fonte do algoritmo desenvolvido foi disponibilizado publicamente na plataforma de versionamento *GitHub* [6], oferecendo à comunidade científica e a profissionais da área um recurso de código aberto para exploração e implementação prática.

II. Fundamentação Teórica

O Aprendizado Federado (AF) estabelece um paradigma de treinamento distribuído no qual o modelo global é aprendido a partir de dados que permanecem nos dispositivos de origem, eliminando a necessidade de centralizar informações potencialmente sensíveis [7]. Esse paradigma segue um ciclo coordenado pelo servidor, inicializando o modelo, selecionando um subconjunto de clientes, recebendo as atualizações treinadas localmente e, por meio de agregação, tipicamente média federada, produz uma nova versão global, que é redistribuída para dar início à rodada seguinte [8]. Ao preservar a privacidade dos usuários e reduzir o tráfego de dados brutos, o AF torna-se particularmente atrativo para sistemas móveis e redes IoT, em que a largura de banda e o consumo energético são restritos [9].

Ainda assim, a heterogeneidade das redes e dos dispositivos impõe desafios múltiplos. Primeiramente, a distribuição *Non-IID* [10] dos dados locais compromete a convergência do modelo quando as atualizações carecem de representatividade estatística; em segundo lugar, a capacidade computacional e energética desigual dos clientes amplifica a variação no tempo de resposta e pode gerar estrangulamento das rodadas de comunicação [11]. Técnicas de regularização, como a penalização L2 empregada neste trabalho, contribuem para estabilizar as atualizações e atenuar o viés introduzido pela divergência de distribuições locais. Adicionalmente, métricas que quantificam a divergência de peso entre modelos sucessivos permitem inferir a relevância de uma atualização antes de transmiti-la ao servidor, evitando *uploads* de baixo impacto e prolongando a autonomia de dispositivos ultrarrestritos.

Dessa forma, este estudo insere-se no esforço de reduzir o custo de comunicação sem sacrificar a acurácia global, pois, ao combinar a regularização L2 com um limiar de divergência de peso, cada cliente decide dinamicamente se enviará ou não o modelo treinado. Essa política local complementa os mecanismos clássicos de agregação, pois atua posteriormente ao treinamento, filtrando envios redundantes e, ao mesmo tempo, preservando a participação estatística de todos os clientes ao longo de múltiplas rodadas.

A. Internet das Coisas

A Internet das Coisas (IoT) descreve um ecossistema composto por dispositivos dotados de sensores e atuadores que capturam o estado do ambiente físico, comunicam-se pela rede e executam ações sem intervenção humana contínua, de forma que esses nós viabilizam aplicações que vão da saúde inteligente e mobilidade urbana à Indústria 5.0 [12], adicionando capilaridade e contexto às soluções digitais [13]–[17]. Para sustentar esse espectro de serviços, o modelo computacional é hierárquico: enquanto a nuvem fornece elasticidade e armazenamento em larga escala, a *edge* aproxima o processamento dos pontos de coleta, diminuindo latência e tráfego de dados [18]–[20].

Contudo, a IoT herda desafios significativos de conectividade. Variações de sinal e interrupções ocasionais são frequentes, especialmente em redes sem fio heterogêneas, o que exige mecanismos de tolerância a falhas nas camadas de aplicação e transporte [21]. Soma-se a isso a diversidade de protocolos como *Message Queuing Telemetry Transport* a *Constrained Application Protocol* e *Hypertext Transfer Protocol* que fragmenta o espaço de desenvolvimento. A inexistência de um protocolo único de ponta a ponta impulsiona esforços para normalização, mas a literatura aponta a ausência de um *framework* unificador capaz de abstrair essas disparidades de forma transparente [22].

A massificação dos dispositivos também impõe desafios de orquestração. Plataformas de gerenciamento devem registrar, provisionar, monitorar e atualizar milhares de nós em tempo real, conciliando restrições de energia, memória e processamento típicas de sensores ultra-restritos [22], [23]. A sus-

tentabilidade passa a ser requisito central: cidades inteligentes, por exemplo, só se concretizam se o consumo energético dos componentes digitais não anular os ganhos sociais pretendidos [24], [25]. Nesse cenário, estudos de medição demonstram que a avaliação de energia precisa combinar modelos baseados em tempo, adequados a dispositivos pouco compartilhados e modelos baseados em capacidade, próprios de servidores e *edge* altamente compartilhados [26]–[28].

Por fim, arquiteturas que deslocam parte do processamento para a borda, como o Aprendizado Federado, emergem como alternativa promissora para equilibrar privacidade, latência e consumo energético. Ainda assim, lacunas permanecem sobre o impacto desse deslocamento no gasto de energia dos nós de *edge* e na divisão ótima de tarefas entre nuvem, *edge* e dispositivos [29]. Dessa forma, compreender a infraestrutura e os requisitos da IoT é essencial para projetar algoritmos federados que respeitem as limitações do domínio e maximizem a eficiência global do sistema.

B. Heterogeneidade dos Dados: Non-IID

McMahan et al. [7] identificaram a distribuição *non-Independent and Identically Distributed - (non-IID)* como o principal obstáculo ao desempenho de algoritmos como o *FedAvg* [10]. A heterogeneidade estatística manifesta-se em múltiplas formas: *feature distribution skew*, quando a distribuição de atributos $P_i(x)$ difere entre clientes mas a relação $P_i(y \mid x)$ permanece inalterada; *label distribution skew*, quando a proporção de rótulos $P_i(y)$ varia enquanto $P_i(x \mid y)$ se mantém semelhante; casos de *mesmo rótulo, diferentes atributos* e de *mesmo atributo, diferentes rótulos*, nos quais a divergência incide, respectivamente, sobre $P_i(x \mid y)$ ou $P_i(y \mid x)$; e, por fim, o *quantity skew*, em que o volume de exemplos $P_i(x, y)$ é muito desigual entre participantes [30].

Essas assimetrias produzem o chamado *client drift*, onde as atualizações locais acabam representando apenas visões parciais do espaço de hipóteses, o que deprecia a acurácia global, retarda a convergência e pode até conduzir à divergência do modelo [30], [31]. Medições empíricas mostram que a perda de desempenho cresce com o desvio entre distribuições e com a proporção de clientes cujos dados cobrem subconjuntos disjuntos das classes de interesse [32].

O estudo [33] agrupa as estratégias de mitigação em quatro grandes eixos. No eixo *data-based*, investiga-se a partilha mínima de exemplos ou rótulos (para "aquecer" o modelo) e técnicas de aumento de dados sintéticos a fim de suavizar vieses locais. No eixo *model-based*, exploram-se regularizações (v.g. L2) ou proximidade ao global), otimizadores adaptativos e esquemas semissíncronos para conter a deriva. O eixo *algorithm-based* inclui abordagens de *meta-learning*, *multi-task learning* e *life-long learning*, que personalizam rapidamente modelos a partir de poucos exemplos locais. Por fim, o eixo *framework-based* recorre a agrupamento de clientes semelhantes, destilação de conhecimento, ou camadas de personalização construídas sobre um núcleo global comum.

Ainda que essas soluções atenuem o problema, o grau de heterogeneidade permanece um fator crítico: quanto maior a distância entre P_i e P_j, maior a penalidade em termos de consumo de comunicação, tempo de treinamento e, sobretudo, qualidade do modelo agregado [33]. Assim, quantificar a divergência estatística, como via variação total, distância de *Wasserstein* ou diferenças de distribuição de classes, torna-se uma etapa importante para orquestrar políticas de seleção de clientes, escalonamento de rodadas ou adaptação de taxa de aprendizado.

O presente estudo dialoga com esse panorama ao propor um limiar de divergência de peso combinado à regularização ℓ_2; tal critério local evita transmissões de baixo impacto em redes IoT ultra-restritas sem replicar a complexidade dos métodos de agrupamento ou seleção reforçada, preservando a equidade de participação ao longo de múltiplas rodadas.

III. Trabalhos Relacionados

Os primeiros trabalhos em AF concentraram-se na redução de latência total por meio de políticas de seleção de clientes. O estudo [34] propõem priorizar participantes cujos dados sejam representativos e cuja velocidade de processamento favoreça a convergência. De forma semelhante, o estudo [35] equilibra capacidade computacional e relevância estatística para minimizar o tempo até atingir um alvo de acurácia. Estratégias baseadas em aprendizado por reforço [36] e meta-heurísticas de arrefecimento simulado [37] estendem esse princípio, treinando agentes que escolhem, a cada rodada, o subconjunto de clientes mais promissor. Em comum, tais abordagens assumem que a eficiência advém de convocar apenas alguns clientes, mas implicam *overhead* de orquestração e podem subutilizar contribuições úteis de nós com baixa prioridade.

Outra linha de pesquisa foca na personalização e agrupamento de clientes. O estudo [38] detecta *botnets* em IoT com aprendizagem federada não supervisionada, explorando a própria heterogeneidade dos dados para formar grupos implícitos. Os estudos [39] e [40] realizam *clustering* explícito, treinando modelos especializados para subconjuntos homogêneos. Já [41] introduz verificações periódicas para realocar clientes ineficientes. Embora eficazes para lidar com distribuições *non-IID*, esses métodos demandam etapas adicionais de agrupamento ou múltiplas instâncias de modelos, elevando o custo computacional no servidor e a carga de sincronização.

Este distingue-se dessas abordagens em três aspectos centrais. Em primeiro lugar, não foram selecionados nem agrupa-

dos clientes; todos permanecem participantes potenciais, mas enviam seus modelos apenas quando a divergência de peso sinaliza contribuição substantiva. Além disso, o critério de decisão é totalmente local, calculado com operações elementares sobre a norma L2 da última camada, evitando qualquer troca extra de metadados, ao contrário dos esquemas que requerem estatísticas de perda ou gradientes para priorização [35], [36]. Por fim, o foco recai sobre dispositivos ultrarrestritos de redes IoT: ao suprimir transmissões quando a variação entre modelos sucessivos fica abaixo do limiar n_0, economizamos mais que o dobro de energia em comparação à estratégia FEV (Estratégia de Minimização de Uploads) baseada em normalização mútua entre cliente e servidor [4], sem degradação perceptível da acurácia global. Assim, contribuímos com uma solução simples, de baixo custo e facilmente embarcável, que complementa os avanços recentes em seleção e personalização sem replicar a sua complexidade.

IV. Avaliação e Análise de Desempenho

Esta seção descreve a configuração dos experimentos, incluindo a utilização da linguagem de programação *Python* [42] e da biblioteca *TensorFlow* com a API *Keras* [43] as quais fornecem ferramentas avançadas para a implementação de redes neurais artificiais. O modelo de rede neural *Multi-Layer Perceptron* (MLP) [44], [45] foi aplicado para processar a base de dados Modified National Institute of Standards and Technology (MNIST) [46], [47], visando avaliar como o algoritmo se adapta à heterogeneidade dos dados e otimiza o desempenho energético em dispositivos IoT ultrarrestritos.

A. Conjunto de Dados

Na configuração do experimento, o conjunto de dados MNIST foi distribuído entre 42 usuários locais, cada um com um número distinto de amostras, que pode variar de centenas a algumas milhares de imagens dos dígitos (0 a 9). Foram utilizados dados *Not Independently and Identically Distributed* (Non-IID) [3], ou seja, os dados não seguem a premissa de serem independentemente e identicamente distribuídos. Em cenários Non-IID, a distribuição dos dados entre os usuários não é uniforme nem balanceada, podendo apresentar padrões que variam de forma significativa entre os indivíduos, ou seja, alguns usuários podem ter acesso predominante a imagens de determinados dígitos, enquanto outros possuem uma representação mais limitada ou até ausente de certas classes. Essa configuração reflete situações mais realistas em aplicações do mundo real, como em sistemas de aprendizado federado, onde os dados coletados localmente por diferentes dispositivos ou usuários frequentemente exibem variabilidades intrínsecas devido a preferências pessoais, condições de coleta ou fatores ambientais.

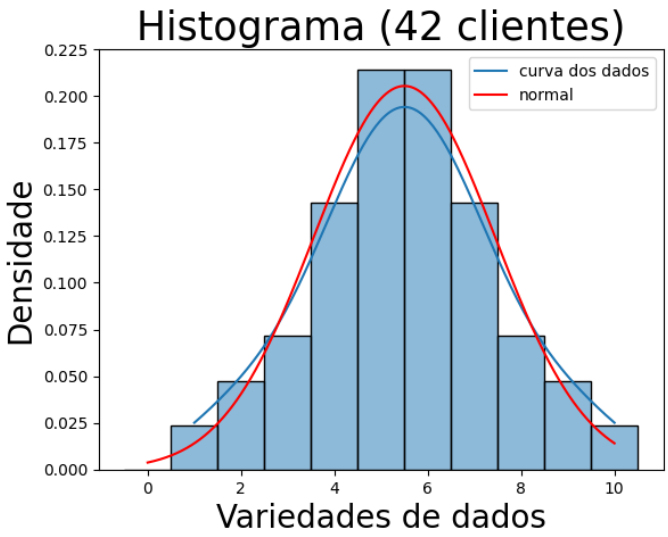

Fig. 1. Distribuição da variedades de dados dos clientes.

A figura 1 ilustra como a variedade de dados (tipos de dígitos que um cliente detém) foi distribuída entre todos os clientes. Como é possível observar, a distribuição real (linha azul) aproxima-se da curva normal correspondente (linha vermelha). Segundo [48], o Teorema Central do Limite (TCL) afirma que, sob certas condições, a soma de um grande número de variáveis aleatórias independentes e identicamente distribuídas tende a uma distribuição normal, independentemente da distribuição original dessas variáveis. Portanto, a distribuição dos clientes nos experimentos busca simular um cenário à medida que o número total de clientes tende ao infinito. Como AF foi desenvolvido para tratar números massivos de clientes, a escolha pela distribuição aproximadamente normal é preferível.

B. Modelo da Rede Neural

Para o desenvolvimento, foram utilizadas as bibliotecas *TensorFlow* e *Keras*. O modelo é composto por três camadas principais: a camada de entrada, que define a forma dos dados com 784 características (por exemplo, uma imagem 28x28 convertida em vetor de 784 características);uma camada oculta com 128 neurônios e função de ativação *ReLU*, que processa os dados; e uma camada de saída com 10 neurônios e a função de ativação *softmax*, totalizando 101.770 parâmetros. Para a otimização, empregou-se o Gradiente Descendente Estocástico (SGD), conhecido por seu desempenho eficiente em grandes conjuntos de dados, conforme destacado em [49]. Além disso, foi utilizada a função de custo *Sparse Categorical Crossentropy*, amplamente aplicada em tarefas de classificação multiclasse.

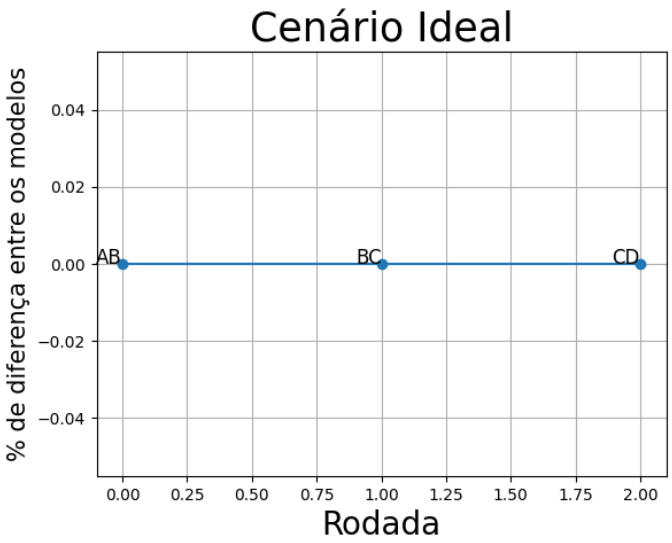

Fig. 2. Representação do Cenário Ideal, onde a decisão do servidor sobre utilizar o módulo enviado pelo DU ou a cópia obsoleta armazenada é irrelevante.

C. Execução

A Função de Controle (FC) busca possibilitar que os DU (Dispositivos de Usuários) poupem energia ao decidirem não enviar seus modelos para o SC (Servidor Central). Além disso, por ser focada em dispositivos hiper-restritos, também deve ser de fácil processamento pelos DU, e aumentar minimamente o armazenamento necessário.

Para aplicar a FC, os DU precisarão somente do seu código implementado, contudo, o SC terá de estar em posse de uma CO (Cópia Obsoleta), que representa uma cópia do último modelo enviado por cada usuário local. Quando solicitados a participar de uma rodada de treinamento, a FC decidirá se o DU enviará modelo para o SC, caso o envio não ocorra, o SC deve usar a CO armazenada desse dispositivo para participar da rodada de treinamento. Portanto, o DU economizará energia ao não enviar seu modelo para o SC.

1) Cenário Ideal: A FC teve como objetivo aproximar os resultados de um cenário ideal, ou seja, um cenário em que é irrelevante para o servidor utilizar tanto o modelo enviado pelo DU quanto a cópia obsoleta armazenada.

Sejam A, B, C e D modelos consecutivos gerados num mesmo DU, caso não haja diferença entre esses modelos, e se o SC estiver em posse de uma CO de A, então, será irrelevante para o SC receber esses modelos do DU, ou usar a CO.

A Figura 2 representa o cenário ideal, sendo uma reta bem definida em zero no plano cartesiano (Rodada × % de diferença entre os modelos), avaliando-se p modelos consecutivos (na Figura, $p = 4$). Como a diferença entre os modelos é de 0%, torna-se irrelevante para o servidor qual modelo (A, B, C ou D) será usado.

Para aproximar os resultados do cenário ideal, foi escolhida uma métrica baseada em operações matemáticas básicas, visando facilitar o processamento pelos DU, e que também fosse capaz de quantificar a linearidade de um conjunto de pontos. Selecionou-se o Coeficiente de Determinação, uma medida estatística que expressa a proporção da variância total da variável dependente explicada por um modelo de regressão linear, indicando o quão bem os dados se ajustam ao modelo. Conforme descrito por [50], ele é calculado como a razão entre a soma dos quadrados explicada pelo modelo (SSR — *Sum of Squares Regression*) e a soma total dos quadrados (SST — *Sum of Squares Total*), utilizando apenas operações matemáticas básicas. Valores próximos de um indicam que o modelo linear explica quase todas as variações nos dados; valores próximos de zero indicam ausência de ajuste significativo.

2) Simbologias: A tabela abaixo concentra as simbologias que serão usadas na fórmula.

TABELA I
PARÂMETROS USADOS NA FC.

Símbolo	Significado
K_m	Modelo local gerado por um DU na rodada de transmissão m que ele participa
$\|\|K_m\|\|$	Normalização L2 da última camada dos pesos da rede neural do modelo K_m
n	Número real positivo
g	Número natural maior ou igual a 2
$\%(K_m, K_{m+1})$	Porcentagem de diferença entre os modelos K_m e K_{m+1}
n_0	Porcentagem de diferença permitida entre dois modelos quaisquer para considerá-los semelhantes. (constante previamente escolhida)
p	Número de modelos analisados para verificar se a porcentagem de diferença entre eles forma uma reta bem definida. (constante previamente escolhida)
$c(x1, x2, x3)$	Coeficiente de determinação da reta formada pelos pontos x1, x2 e x3.

Como dois pontos estão sempre alinhados, para que se obtenha informações úteis, é necessário que o número de pontos, p, seja maior ou igual a três. Além disso, somente a última camada é considerada na avaliação da normalização L2, pois, segundo [51], os pesos dessa camada são mais representativos da performance geral da rede neural em relação ao objetivo final, comparados aos pesos das camadas anteriores. Outrossim, diferença percentual$(K_m, K_{m+1}) = \pm n\%$ que equivale à:

$$\frac{||K_m|| - ||K_{m+1}||}{||K_m||} = \pm n \rightarrow 1 - \frac{||K_{m+1}||}{||K_m||} = \pm n \rightarrow 1 \pm n = \frac{||K_{m+1}||}{||K_m||}$$

3) Estratégia para Minimização de Uploads: A FC busca economizar energia usando poucas informações, para isso, ela é dividida em três condições:

$$I: \quad c(\quad \%(K_m, K_{m+1}), \ldots, \%(K_{m+p-1}, K_{m+p}) \quad) \geq 0.9 \tag{1}$$

$$II: \quad 1 - n_0 \leq \left(\frac{\sum_{x=0}^{p-1} \%(K_{m+x}, K_{m+x+1})}{p} \right) \leq 1 + n_0 \tag{2}$$

$$III: \quad \text{Condição de controle} \tag{3}$$

Se as condições I e II forem ambas satisfeitas, então o cliente não enviará o modelo gerado e, portanto, o servidor central deverá usar a cópia obsoleta. Se, em um mesmo cliente, I e II forem ambos satisfeitos por g vezes consecutivas, então a condição III será usada e o modelo obrigatoriamente será enviado para o servidor.

A condição I busca verificar se a reta formada no plano cartesiano (Rodada) x (% de diferença entre os modelos) para p modelos avaliados é bem definida (coeficiente de determinação maior que 0,9 [52]). No cenário ideal, a reta da condição I seria horizontal em zero. Aqui, um n_0 é selecionado para substituir e, por meio da condição II, é verificado se I está nas proximidades do n_0 escolhido. A condição III apenas assegura que, mesmo que o parâmetro n_0 seja erroneamente escolhido, o modelo ainda convergirá.

Experimentalmente, um n_0 de 5% (0.05) não impactou significativamente o modelo do servidor central e economizou uma quantidade relevante de energia. Valores diferentes de 5%, quando maiores, economizavam mais energia, mas impactavam os resultados do servidor central, por outro lado, quando menores, o inverso ocorria. Em suma, a relação entre energia economizada e impacto nos resultados do servidor central é controlada pelo aumento ou decrescimento do n_0.

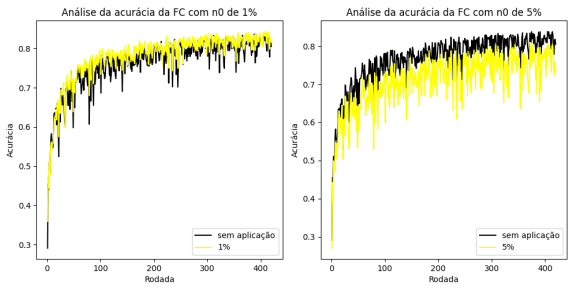

Fig. 3. Comparando resultados da FC

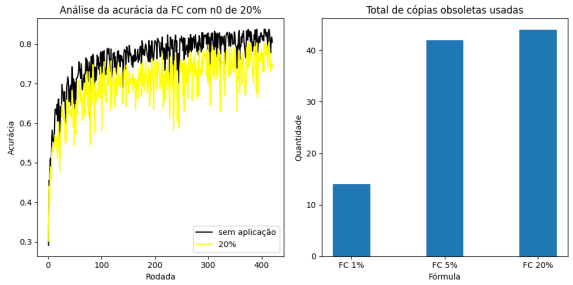

Fig. 4. Comparando resultados da FC

A figura 3 compara os resultados da acurácia centralizada na aplicação da FC (linha amarela), e quando todos os modelos gerados são enviados para o SC (linha preta).

V. Resultados

Esta seção apresenta os resultados de implementação da FC. O código de implementação pode ser conferido em https://github.com/Mateus-de-Almeida/codigoArtigo. Serão usados como comparação os resultados de cenários randômicos e os resultados da aplicação de uma FEV em [4].

Nas Figuras 5 e 6 são comparadas as acurácias centralizadas do servidor central durante as rodadas de comunicação entre dispositivos locais e SC. Foi realizado um total de 420 rodadas, com três clientes participando em cada rodada.

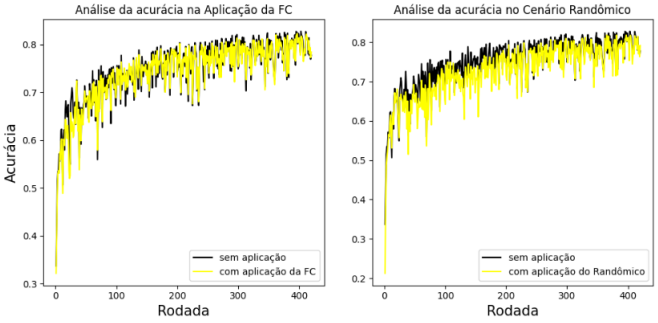

Fig. 5. Resultados FC e Randômico

Na Figura 5, no gráfico à esquerda, é possível observar que a aplicação da FC (linha amarela) não impactou os resultados do SC quando comparada ao cenário em que todos os modelos foram enviados (linha preta). Ou seja, o uso de cópias obsoletas autorizado pela FC não comprometeu o desempenho do servidor central. Já no gráfico à direita, observa-se que um cenário randômico (linha amarela) que economiza a mesma quantidade de energia da FC impacta o SC ao ponto de decrescer a acurácia e, inclusive, alterar o comportamento do treinamento.

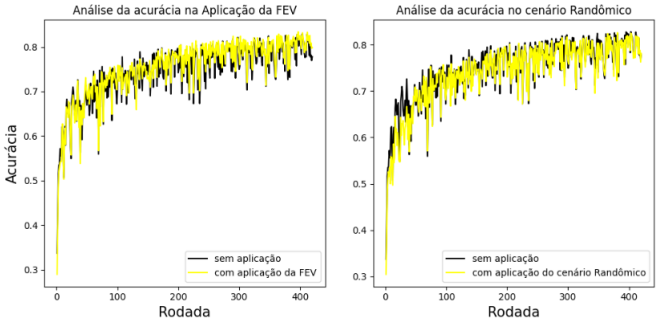

Fig. 6. Resultados FEV e Randômico

Na Figura 6, há os gráficos análogos no cenário da FEV. Contudo, é possível observar que o cenário randômico correspondente não sofreu tanta alteração quanto ao da FC. Logo, as diferenças entre a aplicação da FEV e o cenário randômico correspondente, apesar de existentes, não são grandes.

Embora a FC e a FEV sejam eficazes em seus propósitos, a quantidade de transmissões economizada em cada um foi diferente. Referente aos gráficos gerados da Seção V, a FC economizou a energia de 35 acessos; a FEV, 17 (a FC economizou mais que o dobro da energia economizada pela FEV). Apesar dos experimentos terem sido construídos a fim de simular um cenário real, pautar a eficiência energética entre o FC e o FEV somente neles não é suficiente para concluir qual economizará mais. Contudo, referente à forma como as fórmulas são construídas, considerando o uso de memória, a FC se sai melhor, pois necessita que os DU armazenem somente o valor numérico da porcentagem de diferença entre os p modelos anteriores. Enquanto a FEV necessita que o DU armazene os gradientes do servidor, e a normalização L2 da diferença entre os k modelos anteriores. Já no custo de processamento do DU, a FC apresenta melhor desempenho, pois exige a normalização L2 apenas nos parâmetros do DU, enquanto a FEV requer a normalização L2 tanto no SC quanto no DU. Por fim, quanto ao volume de transmissão entre DU e SC, a FC é mais eficiente, pois demanda menos informações para decidir se o DU enviará seu modelo. Consequentemente, o volume de dados transmitidos pela rede é reduzido.

VI. Conclusão

O Aprendizado Federado (AF) oferece uma abordagem descentralizada para treinar modelos de aprendizado de máquina enquanto preserva a privacidade dos dados. No entanto, a heterogeneidade dos dispositivos e dos dados representa desafios significativos. Este estudo propôs a utilização da divergência de peso como métrica para priorizar atualizações significativas, evitando a transmissão de modelos obsoletos e otimizando o consumo de energia em dispositivos IoT ultrarrestritos.

Os experimentos demonstraram que o algoritmo proposto melhora a eficiência energética e reduz a comunicação, mantendo a qualidade do modelo global. A estratégia evita o envio de modelos irrelevantes ao servidor central, economizando energia e tornando o AF viável para dispositivos hiper-restritos. Além disso, os resultados indicam que a abordagem baseada em divergência de peso é comparável a outras estratégias de economia de energia e tem potencial para ser ampliada.

Trabalhos futuros podem explorar novas métricas, otimizações e métodos de paralelização para aumentar o desempenho em diversos cenários. Estratégias como a variação dinâmica dos parâmetros e algoritmos otimizados para economia de rede e memória são sugeridas para ampliar a aplicação a uma maior variedade de dispositivos. Avanços nessa área são fundamentais para melhorar a eficiência energética e a comunicação, contribuindo para a expansão do aprendizado federado em cenários de dados heterogêneos.

Referências

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, pp. 2787–2805, 10 2010.

[2] H. N. C. Neto, D. M. F. Mattos, and N. C. Fernandes, "Privacidade do usuário em aprendizado colaborativo: Federated learning, da teoria à prática," in *Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSeg)*. Brasil: Sociedade Brasileira de Computação (SBC), 2020, capítulo realizado com recursos do CNPq, CAPES, RNP, FAPERJ e FAPESP (2018/23062-5).

[3] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-iid data: A survey," *Neurocomputing*, vol. 465, 09 2021.

[4] H. Chen, S. Huang, D. Zhang, M. Xiao, M. Skoglund, and H. V. Poor, "Federated learning over wireless iot networks with optimized communication and resources," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 16 592–16 605, 2022.

[5] J. Henkel, S. Pagani, H. Amrouch, L. Bauer, and F. Samie, "Ultra-low power and dependability for iot devices (invited paper for iot technologies)," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*. IEEE, 2017, pp. 954–959.

[6] GitHub, "Github: Where the world builds software," https://github.com, 2025, acessado em: jun. 26, 2025.

[7] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, ser. Proceedings of Machine Learning Research, A. Singh and X. Zhu, Eds., vol. 54. Fort Lauderdale, FL, USA: PMLR, 2017, pp. 1273–1282.

[8] H. N. C. Neto, D. M. F. Mattos, and N. C. Fernandes, "Privacidade do usuário em aprendizado colaborativo: Federated learning, da teoria à prática," in *Minicursos do Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)*, 2020, vol. 20, pp. 142–195.

[9] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[10] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018. [Online]. Available: https://arxiv.org/abs/1806.00582

[11] N.-D. Tran, W. Bao, W. Saad, D. T. Hoang, and D. Niyato, "Federated learning over wireless networks: Optimization model design and analysis," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. IEEE, 2019, pp. 1387–1395.

[12] R. Pereira and N. Santos, "Indústria 5.0: reflexões sobre uma nova abordagem paradigmática para a indústria," *ANPAD. EnANPAD*, pp. 2177–2576, 2022.

[13] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015. [Online]. Available: https://ieeexplore.ieee.org/document/7123563

[14] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, Sep. 2012. [Online]. Available: http://dx.doi.org/10.1016/j.adhoc.2012.02.016

[15] J. Ni, X. Lin, and X. S. Shen, "Toward edge-assisted internet of things: From security and efficiency perspectives," *IEEE Network*, vol. 33, no. 2, pp. 50–57, 2019.

[16] H. Habibzadeh, T. Soyata, B. Kantarci, A. Boukerche, and C. S. Kaptan, "Sensing, communication and security planes: A new challenge for a smart city system design," *Computer Networks*, vol. 144, pp. 163–200, 2018. [Online]. Available: https://doi.org/10.1016/j.comnet.2018.08.001

[17] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of things applications: A systematic review," *Computer Networks*, vol. 148, pp. 241–261, Jan. 2019. [Online]. Available: https://doi.org/10.1016/j.comnet.2018.12.008

[18] H. Elazhary, "Internet of things (iot), mobile cloud, cloudlet, mobile iot, iot cloud, fog, mobile edge, and edge emerging computing paradigms: Disambiguation and research directions," *Journal of Network and Computer Applications*, vol. 128, pp. 105–140, 2019. [Online]. Available: https://doi.org/10.1016/j.jnca.2018.10.021

[19] M. G. Samaila, M. Neto, D. A. B. Fernandes, M. M. Freire, and P. R. M. Inácio, "Challenges of securing internet of things devices: A survey," *Security and Privacy*, vol. 1, no. 2, p. e20, mar 2018. [Online]. Available: http://doi.wiley.com/10.1002/spy2.20

[20] K. Sha, W. Wei, T. A. Yang, Z. Wang, and W. Shi, "On security challenges and open issues in internet of things," *Future Generation Computer Systems*, vol. 83, pp. 326–337, 2018. [Online]. Available: https://doi.org/10.1016/j.future.2018.01.059

[21] F. Montori, L. Bedogni, M. Di Felice, and L. Bononi, "Machine-to-machine wireless communication technologies for the internet of things: Taxonomy, comparison and open issues," *Pervasive and Mobile Computing*, vol. 52, pp. 56–81, 2018. [Online]. Available: https://doi.org/10.1016/j.pmcj.2018.08.002

[22] A. Čolaković and M. Hadžialić, "Internet of things (iot): A review of enabling technologies, challenges, and open research issues," *Computer Networks*, vol. 144, pp. 17–39, 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1389128618305243

[23] Z. Shelby and C. Bormann, *6LoWPAN: The Wireless Embedded Internet*. Hoboken, NJ: Wiley, 2011.

[24] T. Yigitcanlar, M. Kamruzzaman, M. Foth, J. Sabatini-Marques, E. da Costa, and G. Ioppolo, "Can cities become smart without being sustainable? a systematic review of the literature," *Sustainable Cities and Society*, vol. 45, pp. 348–365, feb 2019. [Online]. Available: https://doi.org/10.1016/j.scs.2018.11.033

[25] B. N. Silva, M. Khan, and K. Han, "Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities," *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018. [Online]. Available: https://doi.org/10.1016/j.scs.2018.01.053

[26] A. C. Riekstin and [others], "A survey on metrics and measurement tools for sustainable distributed cloud networks," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 2, pp. 1244–1270, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8226747

[27] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker, "Fog computing may help to save energy in cloud computing," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1728–1739, 2016.

[28] B. Martinez, M. Montón, X. Vilajosana, and J. D. Prades, "The power of models: Modeling power consumption for iot devices," *IEEE Sensors Journal*, vol. 15, no. 10, pp. 5777–5789, 2015.

[29] E. Ahvar, A.-C. Orgerie, and A. Lebre, "Estimating energy consumption of cloud, fog, and edge computing infrastructures," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 2, pp. 277–288, apr 2022. [Online]. Available: https://ieeexplore.ieee.org/document/8668812

[30] K. Hsieh, A. Phanishayee, O. Mutlu, and P. B. Gibbons, "The non-iid data quagmire of decentralized machine learning," *arXiv preprint arXiv:1909.0089*, 2019. [Online]. Available: https://arxiv.org/abs/1909.0089

[31] F. Sattler, S. Wiedemann, K. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3400–3413, 2020. [Online]. Available: https://doi.org/10.1109/TNNLS.2019.2944481

[32] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," *arXiv preprint arXiv:1907.0289*, 2019. [Online]. Available: https://arxiv.org/abs/1907.0289

[33] F. Sattler, K. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[34] B. Luo, Y. Liu, Y. Ji, Y. Cheng, and H. Wang, "Tackling system and statistical heterogeneity for federated learning with adaptive client sampling," *arXiv preprint arXiv:2112.11256*, 2021.

[35] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection," in *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2021, pp. 19–35.

[36] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-iid data with reinforcement learning," in *IEEE INFOCOM*, 2020, pp. 1698–1707.

[37] H. N. Neto, M. R. Oliveira, R. P. Silva, and D. J. Ferreira, "Fedsa: Arrefecimento simulado federado para a aceleração da detecção de intrusão em ambientes colaborativos," in *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, 2021, pp. 280–293.

[38] D. K. Dennis, T. Li, and V. Smith, "Heterogeneity for the win: One-shot federated clustering," *arXiv preprint arXiv:2103.00697*, 2021.

[39] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *arXiv preprint arXiv:2006.04088*, 2020.

[40] F. Sattler, K.-R. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[41] X. Ouyang, J. Li, Z. Wang, Y. Gao, Y. Yu, and X. Qiu, "Clusterfl: A similarity-aware federated learning system for human activity recognition," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2021, pp. 54–66.

[42] G. van Rossum and F. L. Drake, *Python reference manual*. Virginia, USA: PythonLabs, 1995.

[43] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, ..., and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from https://www.tensorflow.org/. [Online]. Available: https://www.tensorflow.org/

[44] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Englewood Cliffs, NJ, USA: Prentice Hall PTR, 1994.

[45] L. B. Almeida, "Multilayer perceptrons," in *Handbook of Neural Computation*. Bristol, UK: IOP Publishing Ltd and Oxford University Press, 1997.

[46] Y. LeCun, C. Cortes, and C. J. C. Burges, "MNIST handwritten digit database," *ATT Labs [Online]*, vol. 2, 2010.

[47] Y. LeCun and C. Cortes, "Mnist handwritten digit database," *ATT Labs [Online]*, vol. 2, 2010. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[48] L. Wasserman, *All of Statistics: A Concise Course in Statistical Inference*, ser. Springer Texts in Statistics. Springer, 2004. [Online]. Available: https://books.google.com.br/books?id=th3fbFI1DaMC

[49] L. Bottou, "Large-scale machine learning with stochastic gradient descent," *Proc. of COMPSTAT*, 01 2010.

[50] N. Draper and H. Smith, *Applied Regression Analysis*, ser. Wiley Series in Probability and Statistics. Wiley, 1998. [Online]. Available: https://books.google.com.br/books?id=d6NsDwAAQBAJ

[51] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.

[52] J. Frost, *Regression Analysis: An Intuitive Guide for Using and Interpreting Linear Models*. Statistics by Jim Publishing, 2020. [Online]. Available: https://books.google.com.br/books?id=1UPzzQEACAAJ