

Deep Q-Network para a Alocação dos Recursos de Comunicação do Aprendizado Federado em Redes IoT sem Fio

Renan R. de Oliveira, Rogério S. e Silva, Leandro A. Freitas e Antonio Oliveira-Jr

Resumo— O Aprendizado Federado (*Federated Learning* - FL) é um paradigma de aprendizado de máquina colaborativo que permite o treinamento de um modelo global sem a exposição dos dados brutos dos dispositivos. Este trabalho investiga o problema da alocação dos recursos de comunicação para tarefas de FL em redes IoT sem fio considerando restrições dos recursos de comunicação e a ocorrência de falhas no processo de treinamento. O algoritmo FL- w_{DQN} é proposto utilizando *Deep Q-Network* (DQN) como técnica de otimização, mostrando-se viável ao alcançar um desempenho equiparável às suas respectivas versões baseadas em Programação Linear.

Palavras-Chave— Aprendizado Federado, Redes IoT sem Fio, Alocação de Recursos, Deep Q-Network.

Abstract— *Federated Learning* is a collaborative machine learning paradigm that allows training a global model without exposing raw data of devices. This work investigates the problem of allocating communication resources for FL tasks in wireless IoT networks considering communication resource constraints and the occurrence of failures in the training process. The FL- w_{DQN} algorithm is proposed using *Deep Q-Network* (DQN) as an optimization technique, proving to be viable by achieving performance comparable to its respective versions based on *Linear Programming*.

Keywords— *Federated Learning*, *Wireless IoT Networks*, *Resource Allocation*, *Deep Q-Network*.

I. INTRODUÇÃO

As aplicações emergentes que utilizam dados gerados por dispositivos da Internet das Coisas (*Internet of Things* - IoT) podem habilitar serviços inteligentes devido aos recentes avanços do Aprendizado de Máquina (*Machine Learning* - ML). Devido à natureza limitada dos recursos de comunicação sem fio, bem como às preocupações com a privacidade, muitas vezes é ineficiente ou impraticável coletar diretamente todos os dados brutos dos dispositivos para uma entidade central [2].

O Aprendizado Federado (*Federated Learning* - FL) [6] foi introduzido como uma abordagem de ML descentralizada que permite que dispositivos treinem de forma colaborativa um modelo compartilhado mantendo os dados privados nos dispositivos. Nesta abordagem, somente os parâmetros dos modelos treinados localmente são compartilhados com o servidor agregador. No contexto das redes sem fio, o FL apresenta

vantagens com relação ao ML centralizado, pois a transmissão dos parâmetros do modelo de ML em vez dos dados de treinamento entre os dispositivos e a Estação Base (*Base Station* - BS) pode economizar energia, recursos de rede e latência da comunicação [12].

Além do mais, a otimização dos recursos de comunicação melhoram a eficiência do FL em redes sem fio [1]. Dessa forma, as técnicas de ML baseadas em Aprendizado por Reforço (*Reinforcement Learning* - RL) permitem a otimização dos recursos com base na interação de um agente com o ambiente e no aprendizado de ações que maximizam uma recompensa ao longo do tempo. Uma vantagem do RL é sua capacidade de encontrar soluções viáveis em vez de limitar-se na busca de soluções ótimas, como é o caso dos métodos de otimização determinísticos baseados em Programação Linear.

Neste contexto, este trabalho propõe o algoritmo FL- w_{DQN} ¹, cujo objetivo é prover uma implementação baseada em *Deep Q-Network* (DQN) para a otimização da alocação dos recursos de comunicação para tarefas de FL em redes IoT sem fio. Neste caso, devido à grande quantidade de pares {estado, ação}, o método DQN emprega uma rede neural para representar a política. O problema do escalonamento de recursos é tratado como um problema de designação para a alocação de Blocos de Recursos (*Resource Blocks* - RB) de *uplink*, cujo objetivo é minimizar a taxa de erro de transmissão de pacotes atendendo uma política que determina requisitos de latência de comunicação, consumo energético e taxa de erro de pacote.

As contribuições centrais deste artigo são elencadas a seguir: (i) apresentam-se as formulações matemáticas que subsidiaram a definição do problema de otimização e a simulação da proposta do algoritmo FL- w_{DQN} ; (ii) discute-se FL- w_{DQN} que minimiza a taxa de erro de transmissão de pacotes; (iii) discute-se os resultados da implementação de versões dos algoritmos FedAvg [6] e POC [13], denominados neste trabalho como FedAvg- w_{DQN} e POC- w_{DQN} que incorporam o conhecimento do meio de transmissão com base em FL- w_{DQN} ; (iv) apresenta-se resultados por meio de simulações que mostram que as versões dos algoritmos que utilizam FL- w_{DQN} são equiparáveis às suas respectivas versões implementadas com base na técnica de Programação Linear Inteira Mista (*Mixed-Integer Linear Programming* - MILP).

Para além desta seção introdutória, o restante deste artigo está organizado conforme descrito a seguir. A Seção II discute

Renan R. de Oliveira, Universidade Federal de Goiás, Goiânia-GO e Instituto Federal de Goiás, Goiânia-GO, e-mail: renan.rodrigues@ifg.edu.br. Rogério S. e Silva, Universidade Federal de Goiás, Goiânia-GO e Instituto Federal de Goiás, Inhumas-GO, e-mail: rogerio.sousa@ifg.edu.br. Leandro A. Freitas, Instituto Federal de Goiás, Inhumas-GO, e-mail: leandro.freitas@ifg.edu.br. Antonio Oliveira-Jr, Universidade Federal de Goiás, Goiânia-GO e Fraunhofer Portugal AICOS, Porto-Portugal, e-mail: antoniojr@ufg.br.

¹Disponível em <https://github.com/LABORA-INF-UFG/FL-wDQN>

os trabalhos relacionados. A Seção III apresenta os modelos matemáticos que foram utilizados na simulação dos algoritmos de FL em redes IoT sem fio. A Seção IV descreve a formulação do problema de otimização e a proposta do algoritmo FL- w_{DQN} . A Seção V discute os resultados e a análise da simulação. Por fim, a Seção VI apresenta as considerações finais e indica as orientações para os trabalhos futuros.

II. TRABALHOS RELACIONADOS

Na formulação inicial do FL proposta por [6], os autores introduziram o conceito de FL e apresentaram o desempenho do algoritmo *Federated Averaging* (FedAvg). Em [13], os autores apresentam o algoritmo *Power-Of-Choice* (POC), que seleciona os dispositivos para a próxima rodada de comunicação com base nos maiores valores da função de perda local. Entretanto, os trabalhos de [6] e [13] não apresentaram resultados que pudessem destacar a eficácia da proposta em um cenário realista de redes sem fio.

O artigo de [10] apontam os desafios do FL devido a natureza dinâmica das redes sem fio. Em [3], os autores apresentam um modelo de FL em redes sem fio considerando características de computação e comunicação. Em [1], é apresentado um modelo de FL em redes sem fio como um problema de otimização que visa minimizar a perda de treinamento do modelo global. De forma semelhante, o estudo de [2] investigou a otimização de recursos de comunicação para o FL. Contudo, as pesquisas de [1] e [2] não abordam de maneira abrangente a escalabilidade para ambientes de redes IoT sem fio, uma vez que os resultados foram obtidos com base na iteração de poucos dispositivos.

Motivado pelos desafios do FL no contexto das redes sem fio, este trabalho propõe o algoritmo FL- w_{DQN} baseado em DQN para otimização dos recursos de comunicação, envolvendo uma densidade de dispositivos ligeiramente superior em comparação com os trabalhos citados da literatura.

III. MODELO DO SISTEMA

A. Modelo de Rede para FL

Considere uma rede IoT sem fio com uma BS conectada diretamente a um servidor agregador de FL com N dispositivos. Os dispositivos possuem capacidade para coletar dados e treinar um modelo local para uma determinada tarefa de FL. A ideia do FL é realizar a agregação de um modelo global de ML ao longo de várias rodadas de comunicação com base nos parâmetros dos modelos treinados localmente em cada dispositivo minimizando a função de perda $f(w_{global})$. No início de cada rodada, a BS deve selecionar uma fração de N dispositivos e utilizar os canais de *downlink* para transmitir o estado atual do modelo global para cada dispositivo.

Cada participante treina um modelo local w_i durante E épocas locais. Após o treinamento, cada dispositivo utiliza um canal de *uplink* para enviar w_i para a BS conectada a um servidor agregador de parâmetros. Em seguida, os modelos locais são recebidos e agregados pelo servidor, gerando o estado atual w_{t+1} do modelo global. De acordo com [5], o processo de treinamento distribuído deve ser repetido por várias rodadas até que a convergência seja alcançada ou o desempenho do modelo alcance um critério de parada.

B. Modelo de Comunicação

Considere a técnica de acesso múltiplo por divisão de frequência ortogonal (OFDMA) para o *uplink*, onde cada dispositivo ocupa um RB. De acordo com [1], a taxa de *uplink* do dispositivo i transmitindo os parâmetros do modelo w_i para a BS pode ser formulada como

$$c_i^U(r_i, P_i) = \sum_{n=1}^R r_{i,n} B_n^U \mathbb{E} \left(\log_2 \left(1 + \frac{P_i h_i}{I_n + B_n^U N_0} \right) \right), \quad (1)$$

onde $r_i = [r_{i,1}, \dots, r_{i,R}]$ é o vetor de alocação de RBs, $r_{i,n} \in \{0, 1\}$ e $\sum_{n=1}^R r_{i,n} = 1$, com $r_{i,n} = 1$ indicando que o RB n está alocado para o dispositivo i . O ganho do canal é dado por $h_i = o_i d_i^{-\alpha}$, onde d_i é a distância entre o dispositivo i e a BS, o_i é o parâmetro de desvanecimento de Rayleigh e α é um expoente que afeta como o ganho do canal varia com a distância. A taxa de dados de *downlink* alcançada pela BS ao transmitir os parâmetros do modelo global é dada por

$$c_i^D = B^D \mathbb{E} \left(\log_2 \left(1 + \frac{P_B h_i}{I^D + B^D N_0} \right) \right). \quad (2)$$

Assume-se que os modelos de FL são transmitidos por meio de um único pacote. Assim, o atraso de transmissão entre um dispositivo i e a BS no *uplink* e *downlink* podem ser respectivamente formulados como

$$l_i^U(r_i, P_i) = \frac{S_{pkt}^U}{c_i^U(r_i, P_i)}, \quad (3) \quad l_i^D = \frac{S_{pkt}^D}{c_i^D}, \quad (4)$$

onde S_{pkt}^U é o tamanho do pacote de *uplink* e S_{pkt}^D é o tamanho do pacote de *downlink*.

Considera-se que a BS não solicitará aos dispositivos o reenvio de modelos quando estes forem recebidos com erros. Neste caso, conforme apresentado por [1], a taxa de erro de transmissão de pacote do *uplink* é dado por

$$q_i^U(r_i, P_i) = \sum_{n=1}^R r_{i,n} \mathbb{E} \left(1 - \exp \left(-\frac{m(I_n + B^U N_0)}{P_i h_i} \right) \right). \quad (5)$$

C. Modelo de Consumo de Energia

A demanda energética da BS não é considerada, uma vez que esta normalmente possui um fornecimento contínuo de energia. Baseado em [1], o modelo de consumo de energia dos dispositivos para o treinamento e transmissão do modelo w_i podem ser formulados como

$$e_i^T = \zeta \omega_i \vartheta^2 Z(w_i), \quad (6) \quad e_i^U = P_i l_i^U, \quad (7)$$

onde ϑ , ω_i e ζ referem-se, respectivamente, a frequência do *clock*, o número de ciclos da unidade central de processamento e o coeficiente de consumo de energia de cada dispositivo. Dessa forma, em uma rodada de comunicação, a computação da energia consumida pelo dispositivo i é dada por

$$e_i^R = e_i^T + e_i^U. \quad (8)$$

IV. DQN PARA O ESCALONAMENTO DOS RECURSOS DE COMUNICAÇÃO

A. Formulação do Problema de Otimização

Considere uma matriz binária c como um indicador de atribuição de um dispositivo i no RB j , cujo objetivo é

minimizar a taxa de erro de transmissão de pacotes de *uplink*. Dessa forma, o problema do escalonamento dos recursos de comunicação para tarefas de FL pode ser formulado como

$$\min \sum_{i=1}^{n_p} \sum_{j=1}^R q_i^U c_{ij} - \lambda \sum_{i=1}^{n_p} \sum_{j=1}^R c_{ij}, \quad (9)$$

$$\text{s.a.} \sum_{i=1}^{n_p} c_{ij} = 1 \quad \forall j \in R, \quad (9a)$$

$$\sum_{j=1}^R c_{ij} = 1 \quad \forall i \in n_p, \quad (9b)$$

$$l_i^U \leq \gamma_T, \quad (9c)$$

$$e_i^U \leq \gamma_E, \quad (9d)$$

$$q_i^U \leq \gamma_Q, \quad (9e)$$

$$\sum c_{ij} \leq n_f, \quad (9f)$$

$$c_{ij} \in \{0, 1\}, \quad (9g)$$

onde λ é um peso que controla a importância da maximização da quantidade de dispositivos para cada rodada de comunicação. A restrição (9a) garante que cada RB é alocado para um único dispositivo e a restrição (9b) garante que cada dispositivo é atribuído a exatamente um RB. As restrições (9c), (9d) e (9e) estabelecem restrições de latência, consumo energético e da taxa de erro de pacote para a transmissão dos modelos dos dispositivos para a BS. A restrição (9f) garante que o número final de dispositivos é no máximo n_f e a restrição (9g) define que c é uma matriz binária, onde $c_{ij} = 1$ indica que o RB j foi alocado para o dispositivo i e $c_{ij} = 0$ indica o contrário.

B. Proposta do Algoritmo FL- w_{DQN}

O algoritmo 1 apresenta um visão geral de FL- w_{DQN} em redes IoT sem fio. Na linha 5 do Algoritmo 1, observa-se que no início de cada rodada de comunicação, o agente DQN é treinado para aprender uma política de alocação de recursos considerando γ_T , γ_E e γ_Q . A política é a estratégia que define como o agente deve escolher suas ações em função dos estados do ambiente. O estado é a informação recebida pelo agente para determinar qual ação deve ser tomada. Após cada ação, o agente é levado a um novo estado e recebe uma recompensa que serve para avaliar a eficácia da ação executada com base no valor da função objetivo. A seguir são apresentados alguns detalhes relevantes da proposta do algoritmo FL- w_{DQN} :

- Espaço de Estados: O espaço de estados abrange todas as permutações para a alocação de RBs para os dispositivos em cada rodada de comunicação. Cada estado é definido como uma tupla $S_k = (k, v)$, onde k é o índice do estado e $v = [r_{i,1}, \dots, r_{i,R}]$ é uma possível configuração de alocação de recursos com $r_{i,r}$ indicando o índice do dispositivo i atribuído ao RB r .
- Espaço de Ações: As ações são definidas com base na mudança da configuração do vetor de alocação de RBs, utilizando quatro estratégias: operação de incremento e

Algoritmo 1: FL- w_{DQN} em Redes IoT sem Fio

```

1 Servidor:
2   inicialização de  $w_0$ 
3   para cada rodada  $t = 1, 2, \dots$  faça
4      $S_t \leftarrow$  (selecione  $m$  dispositivos)
5      $\triangleright$ Gerando a política de alocação de recursos
       Treine o agente com DQN para resolver o
       problema de otimização da Equação (9)
6      $\triangleright$ Alocando os RBs de uplink
       Defina  $c$  com  $n_f \leq n_p$  dispositivos com base na
       política construída na fase de treinamento
7     para cada  $c_{ij} = 1$  em paralelo faça
8        $w_{t+1}^i \leftarrow$  ClienteUpdate( $i, w_t$ )
9        $m_t \leftarrow \sum_{i \in S_t} n_i$     $w_{t+1} \leftarrow \sum_{i \in S_t} \frac{n_i}{m_t} w_{t+1}^i$ 
10  ClienteUpdate( $i, w$ ):  $\triangleright$  Para cada dispositivo  $i$ 
11     $\mathcal{B} \leftarrow$  (divisão de  $\mathcal{P}_i$  em lotes de tamanho  $\mathcal{B}$ )
12    para cada época local  $j$  de 1 até  $E$  faça
13      para cada  $b \in \mathcal{B}$  faça
14         $w \leftarrow w - \eta \nabla \ell(w; b)$ 
15    retorne  $w$  para o servidor
    
```

decremento do índice do estado; operação de rotação do vetor de alocação de RBs para esquerda e para a direita.

- Recompensas: Após uma ação que leva o agente a um novo estado, a recompensa é calculada considerando o valor dado pela Equação 9. O agente é penalizado (com um valor elevado de magnitude positiva) caso escolha uma ação para um estado que viola as restrições da formulação do problema.

Durante a fase de treinamento, o agente executa uma sequência de ações conforme aprende uma política de alocação de recursos, repetindo esse procedimento por várias iterações em cada episódio. Em seguida, na Linha 6 do Algoritmo 1, define-se o vetor c de alocação de recursos percorrendo a política definida na fase de treinamento. Inicialmente, determina-se um estado qualquer da política, e logo em seguida, a política é percorrida por $n = 5$ vezes de forma não-determinística (seguindo a política em 80% dos casos em 20% dos casos selecionando outro estado da política). Enquanto isso, é armazenando o estado que minimiza o valor da função objetivo.

V. AVALIAÇÃO E ANÁLISE DE DESEMPENHO

Considere uma rede IoT sem fio atendendo uma área circular com raio r de 500 metros com uma BS no centro. A BS está diretamente associada a um servidor agregador, com $N = 100$ dispositivos conectados para uma tarefa de FL. Os dispositivos são distribuídos de maneira uniforme e aleatória com distâncias entre 100 e 500 metros da BS. A largura de banda de *uplink* de cada RB é de 1 MHz com a atribuição de uma interferência distinta e incremental. Cada dispositivo é configurado para transmitir os modelos locais para a BS com a potência de 0.01 W. A largura de banda do *downlink* é de 20 MHz, onde a BS transmite os modelos globais para os dispositivos com uma potência de 1 W. A

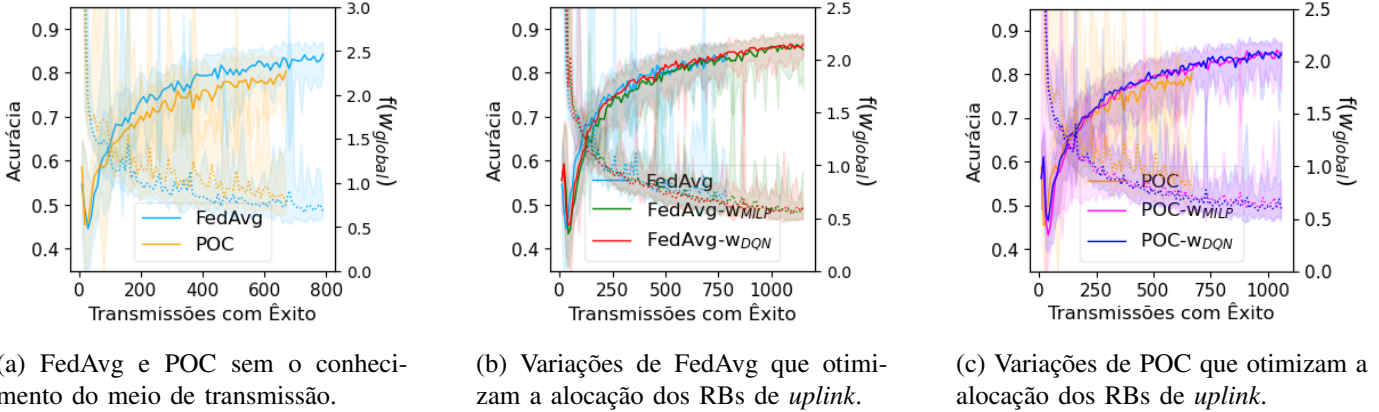


Fig. 1: Evolução da acurácia e $f(w_{global})$, onde os algoritmos FedAvg- w_{DQN} e POC- w_{DQN} incorporam o conhecimento do meio de transmissão utilizando o algoritmo FL- w_{DQN} .

modelagem de h_k incorpora um efeito de desvanecimento que indica que o ganho do canal diminui conforme aumenta a distância dos dispositivos em relação a BS. Conforme [1], outros parâmetros incluem: $\alpha = 2$, $N_0 = -174$ dBm/Hz, $m = 0.023$ dB, $\vartheta = 10^9$, $\zeta = 10^{-27}$ e $\omega_i = 40$.

As tarefas de FL consideram problemas de classificação de imagens utilizando o conjunto de dados de *benchmark* utilizado em pesquisas de FL denominado MNIST, conforme os trabalhos de [6], [4], [11], [7], [1], [8] e [2]. Diferentemente das referências citadas, este trabalho utiliza uma variação do MNIST que tornam os dados heterogêneos como uma característica inerente das redes IoT sem fio. Dessa forma, o conjunto de dados foi dividido em 10 subconjuntos com amostras do mesmo rótulo, utilizando 75% das amostras para o conjunto de treinamento e 25% para o conjunto de teste. Em seguida, cada dispositivo recebeu uma partição de treinamento e teste, onde: 90% das amostras pertencem ao mesmo rótulo e os 10% restantes pertencem igualmente aos demais rótulos; cada imagem é rotacionada em até 45° em sentido horário ou anti-horário; a quantidade final de dados é dada por um fator entre $[0.25, 1]$ da partição inicial.

Neste trabalho, as partições do MNIST foram nomeadas como NIID R-MNIST. Para a tarefa de classificação do conjunto de dados, utilizou-se uma arquitetura de rede neural do tipo *Multi-Layer Perceptron* (MLP) com 101.770 parâmetros treináveis. Cada modelo foi treinado por uma época de treinamento local, definindo $\gamma_T = 200$ ms como requisito de atraso, $\gamma_E = 0.0025$ J como requisito de consumo energético e $\gamma_Q = 0.3$ como a taxa permitida de erros de pacotes.

A. Resultados da Simulação

Esta seção discute a implementação das variações dos algoritmos FedAvg e POC, denominadas neste trabalho como FedAvg- w_{DQN} e POC- w_{DQN} , incorporando o conhecimento da rede sem fio utilizando o algoritmo FL- w_{DQN} para a alocação dos RBs de *uplink*. Para propósito de comparação, também implementou-se as versões FedAvg- w_{MILP} e POC- w_{MILP} que incorporam o conhecimento do meio de transmissão utilizando um algoritmo baseado em MILP.

Em razão da natureza estocástica inerente à simulação,

cada algoritmo foi executado por 15 vezes com o objetivo de obter estatísticas relacionadas ao desempenho médio. Os algoritmos foram executados por 200 rodadas de comunicação, onde no máximo $n_f = 10$ dispositivos foram selecionados em cada rodada, cabendo aos otimizadores a realização do escalonamento da mesma quantidade de RBs de *uplink*.

A Figura 1 apresenta a evolução das acurácias e $f(w_{global})$ dos algoritmos de FL considerando a quantidade de transmissões com êxito. As curvas sólidas representam o comportamento médio e as regiões sombreadas correspondem aos valores de mínimo e máximo. Observa-se que as acurácias aumentam na medida do crescimento das transmissões com êxito, permitindo que o modelo global se beneficie das informações adquiridas nos padrões de dados dos dispositivos.

De forma análoga, observa-se que na medida em que o número de transmissões com êxito aumentam os valores de $f(w_{global})$ diminuem. Além do mais, nota-se na Figura 1a que os algoritmos FedAvg e POC são incapazes de transmitir uma quantidade significativa de modelos locais com êxito devido ao desconhecimento das condições do meio de transmissão. De forma análoga, as Figuras 1b e 1c apresentam a evolução das acurácias e $f(w_{global})$ dos algoritmos que incorporaram o conhecimento do meio de transmissão para a otimização da atribuição dos RBs de *uplink*.

Neste caso, nota-se que os algoritmos que utilizam FL- w_{DQN} são capazes de transmitir uma quantidade significativamente maior de modelos locais com êxito em comparação com as suas versões originais que desconhecem as condições do meio de transmissão. Além do mais, FedAvg- w_{DQN} e POC- w_{DQN} também são capazes de transmitir uma quantidade equiparável de modelos com êxito com relação às suas respectivas versões baseados em MILP. Portanto, os algoritmos que incorporaram o conhecimento do meio de transmissão aumentaram a quantidade de transmissões com êxito, e como consequência, tendem a melhorar a acurácia do modelo global.

Após 200 rodadas de comunicação, o melhor desempenho dos algoritmos FedAvg, FedAvg- w_{MILP} e FedAvg- w_{DQN} , apresentaram, respectivamente, uma acurácia no valor de 84.55, 86.90 e 86.59. Da mesma forma, os algoritmos POC, POC- w_{MILP} e POC- w_{DQN} , apresentaram, respectivamente,

uma acurácia no valor de 80.46, 85.57 e 85.37.

A Figura 2 apresenta a soma dos tempos de ocupação dos RBs para o treinamento e transmissão dos modelos locais em cada rodada de comunicação. Neste caso, observa-se que os algoritmos baseados em FL- w_{DQN} minimizam a utilização dos recursos ao reduzir o tempo de ocupação dos RBs de forma similar aos algoritmos baseados em MILP.

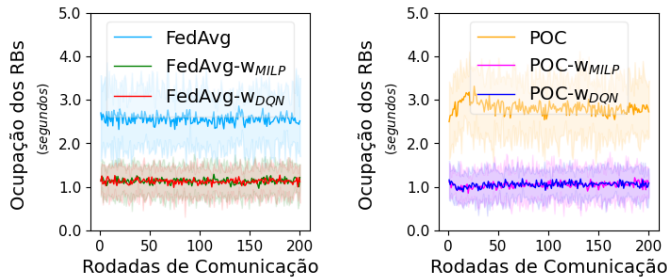


Fig. 2: Tempo de ocupação dos RBs de *uplink*.

Como FedAvg e POC desconhecem o meio de transmissão, não é possível a realização de ações visando o êxito das transmissões. Portanto, FedAvg e POC determinam que todos os dispositivos selecionados para cada rodada de comunicação treinem e transmitam seus modelos. Neste caso, os RBs de *uplink* são ocupados pelo tempo necessário para a realização destas tarefas, mesmo que isto esteja em desacordo com γ_T , γ_E e γ_Q . A Figura 3 destaca que o número de transmissões com êxito de FedAvg e POC é inferior às suas respectivas versões que otimizam a alocação dos recursos de comunicação considerando um intervalo de confiança de 95%.

O custo energético dos algoritmos também são apresentados na Figura 3, onde as barras sólidas denotam o custo energético total e as barras com preenchimento riscado apresentam o custo energético dos erros de transmissão. Nota-se que FedAvg e POC consomem a maior parte dos recursos energéticos no treinamento e na transmissão de modelos que resultam em erros de transmissão. Por outro lado, os algoritmos que utilizam FL- w_{DQN} demonstram eficiência na alocação dos recursos de forma mais assertiva para o uso dos recursos energéticos, bem como, são equiparáveis às suas respectivas versões baseadas em MILP.

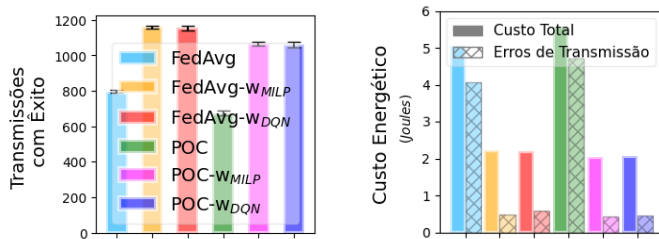


Fig. 3: Total de transmissões com êxito e custo energético.

Portanto, nota-se que os resultados estão em conformidade com a função objetivo da Equação (9), onde as restrições (9c) e (9d) garantem que somente n_f dispositivos que atendem os requisitos γ_T , γ_E e γ_Q devem treinar e transmitir os seus modelos. Em comparação com suas respectivas versões

sem o conhecimento do meio de transmissão, destaca-se que FedAvg- w_{DQN} e POC- w_{DQN} apresentam, respectivamente, uma eficiência energética total de 43.57% e 36.98%.

VI. CONCLUSÕES

Este trabalho apresentou a formulação do algoritmo FL- w_{DQN} que otimiza a alocação dos recursos de comunicação em redes IoT sem fio para tarefas de FL. Os resultados mostram que as versões dos algoritmos baseados em FL- w_{DQN} transmitiram uma quantidade significativamente maior de modelos locais com êxito, melhoraram a acurácia do modelo global, reduziram o tempo de ocupação dos RBs e demonstram eficiência no uso dos recursos energéticos em comparação com suas respectivas versões sem o conhecimento do meio de transmissão. Além do mais, os algoritmos baseados em FL- w_{DQN} mostram-se viáveis pois alcançam um desempenho similar aos algoritmos baseados em MILP que sempre prioriza a busca pela alocação ótima dos recursos de comunicação. Ressalta-se que o código-fonte de FL- w_{DQN} está disponível para permitir a reprodução e validação dos resultados. Para trabalhos futuros, pretende-se utilizar o simulador de rede NS3 para a experimentação de soluções atuais de FL e a proposição de novos algoritmos para a otimização de diferentes arquiteturas de FL em redes sem fio.

REFERÊNCIAS

- [1] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A Joint Learning and Communications Framework for Federated Learning Over Wireless Networks", *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 269-283, 2021.
- [2] 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. 16592-16605, 2022.
- [3] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated Learning over Wireless Networks: Optimization Model Design and Analysis", in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1387-1395, 2019.
- [4] Y. Amannejad, "Building and Evaluating Federated Models for Edge Computing", in *2020 16th International Conference on Network and Service Management (CNSM)*, pp. 1-5, 2020.
- [5] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection", *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, pp. 1-1, 2021.
- [6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data", *arXiv*, arXiv:1602.05629, 2016.
- [7] G. Zhu, Y. Wang, and K. Huang, "Broadband Analog Aggregation for Low-Latency Federated Edge Learning", *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491-506, 2020.
- [8] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated Learning with Non-IID Data", *arXiv*, arXiv:1806.00582, 2022.
- [9] M. Asad, A. Moustafa, T. Ito, and M. Aslam, "Evaluating the Communication Efficiency in Federated Learning Algorithms", *CoRR*, arXiv:2004.02738, 2020.
- [10] X. Cao, T. Başar, S. Diggavi, Y. C. Eldar, K. B. Letaief, H. V. Poor, and J. Zhang, "Communication-Efficient Distributed Learning: An Overview", *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 4, pp. 851-873, 2023.
- [11] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, T. Parcollet, and N. D. Lane, "Flower: A Friendly Federated Learning Research Framework", *CoRR*, arXiv:2007.14390, 2020.
- [12] Z. Yang, M. Chen, K. Wong, H. V. Poor, and S. Cui, "Federated Learning for 6G: Applications, Challenges, and Opportunities", *Engineering*, vol. 8, pp. 33-41, 2022.
- [13] Y. J. Cho, J. Wang, and G. Joshi, "Client Selection in Federated Learning: Convergence Analysis and Power-of-Choice Selection Strategies", *CoRR*, arXiv:2010.01243, 2020.