Algoritmo Baseado em Aprendizado por Reforço e Modelo Markoviano para Alocação de Recursos em um Sistema Internet das Coisas Cognitivo

Matheus Matos Vasconcelos, Álisson Assis Cardoso, Flávio Henrique Teles Vieira

***Resumo*— Este artigo propõe a utilização de um algoritmo de aprendizado por reforço para controlar a transmissão de pacotes de múltiplos dispositivos em um sistema de comunicação sem fio baseado no conceito de Internet das Coisas (IdC) Cognitivo. A abordagem proposta consiste em adotar uma cadeia de Markov para modelar os estados do sistema de comunicação e suas transições, fornecendo os parâmetros necessários para determinar ações para o sistema através de um algoritmo *Q-Learning*. O artigo também apresenta uma avaliação do desempenho do algoritmo desenvolvido em comparação aos de alguns algoritmos de escalonamento em termos de vários parâmetros, tais como: função de utilidade, vazão, taxa de perda de pacotes, etc.**

***Palavras-Chave*— Aprendizado por Reforço, Cadeia de Markov, Escalonamento, Internet das Coisas (IdC).**

***Abstract*— This article proposes a utilization of a reinforcement learning algorithm to control the packet transmission of multiple devices of a Cognitive Internet of Things (IoT) wireless communication system. The proposed approach consists of adopting a Markov chain to model the states of the communication system and its transitions, providing the required parameters to determine actions to the system using a Q-Learning algorithm. The article also presents a performance evaluation of the developed algorithm in comparison to some scheduling algorithms in terms of: utility function, flow rate, packet loss rate, etc.**

***Keywords*— Reinforcement Learning, Scheduling, Markov Chain, Internet of Things (IoT).**

I. Introdução

A Inteligência Artificial (IA) vem sofrendo um crescente aumento em seu uso em diferentes ramos da ciência uma vez que os problemas vêm se tornando cada vez mais complexos e demandando soluções menos restritivas que se adaptem a natureza não trivial das dificuldades modernas [1], [2]. Nesses cenários, o emprego de Inteligência Artificial é ainda mais desejável visto que a mesma possibilita que sistemas sejam capazes de aprender e tomar decisões onde não há soluções ótimas claras.

Os algoritmos inteligentes podem facilitar o tratamento de grandes quantidades de dados, podem aumentar a velocidade de análise e permite que processos complexos sejam automatizados [1], [3]. Assim, surge a oportunidade do uso de IA como uma ferramenta de coordenação de ações em cenários IdC cognitivos, onde dispositivos e sensores inteligentes exigem uma certa flexibilidade do sistema por conta da dinâmica de integração, resultando em cenários mais complexos e com uma ampla variedade de aplicações.

Algoritmos baseados em técnicas de aprendizado de máquina possuem capacidade de aprender e se adaptar, permitindo a otimização dos recursos do sistema. No caso de um cenário IdC cognitivo, por exemplo, a aplicação de tais algoritmos pode proporcionar aumento de vazão de dados enquanto se procura reduzir o custo total de transmissão, melhorando assim a qualidade de serviço e a eficiência do sistema [3], [2]. No aprendizado por reforço são avaliadas as ações possíveis a serem tomadas, permitindo determinar um curso de ações para cada estado do sistema levando em consideração as recompensas obtidas para cada ação.

Neste trabalho, propõe-se a utilização de um algoritmo de aprendizagem por reforço que faz uso de um modelo de Markov para descrever as probabilidades de transições dos estados de um sistema IdC cognitivo e determinar uma política de ações que uma estação base deverá tomar para aumentar a utilidade do sistema. Em outras palavras, a abordagem proposta objetiva aumentar o valor da razão entre a quantidade de pacotes transmitidos e a potência consumida para a transmissão.

Em resumo, este trabalho apresenta as seguintes contribuições:

1) Modelagem de um sistema IdC cognitivo (que opera com compartilhamento de M canais de transmissão) por meio de uma cadeia de Markov, que permite determinar os estados do sistema e as transições dos mesmos e calcular o valor da utilidade de cada ação possível tomada em um determinado estado. Diferente de outros trabalhos da literatura [4], [5], o sistema é capaz de transmitir pacotes de mais de um dispositivo ao mesmo tempo.
2) Proposta de um algoritmo de aprendizado por reforço para testar diferentes políticas de ações e determinar alocação de recursos de forma a otimizar uma função de utilidade do sistema.

O artigo está organizado da seguinte maneira, as seções II e III tratam da modelagem do sistema, a seção IV descreve o algoritmo de alocação de recursos proposto, os outros algoritmos de escalonamento considerados são descritos na seção V e, finalmente, a seção VI apresenta os resultados das simulações e uma análise dos resultados.

Matheus Matos Vasconcelos, Álisson Assis Cardoso, Flávio Henrique Teles Vieira, Escola de Engenharia Elétrica, Mecânica e de Computação, Universidade Federal de Goiás, Goiânia-GO, e-mail: mmatosvaconcelos@gmail.com, alsnac@gmail.com, flaviohtv@gmail.com. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

II. Modelagem do Sistema

O sistema IdC cognitivo mostrado na Figura 1 consiste em uma estação base com M canais de transmissão disponíveis para K dispositivos que transmitem pacotes de dados. O tempo é discretizado em intervalos iguais e em cada intervalo de tempo, pacotes chegam à cada dispositivo que os transmite caso haja um canal disponível. O sistema é capaz de transmitir pacotes de mais de um dispositivo ao mesmo tempo. Cada dispositivo possui um *buffer* de tamanho L para armazenar pacotes que não foram transmitidos, onde os pacotes chegam obedecendo uma distribuição de Poisson com a taxa de chegada λ e são transmitidos por um dos M canais com uma taxa de codificação V, não há transmissão quando a qualidade do canal é mínima. Quando pacotes chegam ao *buffer* de um dispositivo que está cheio, há perda de pacotes.

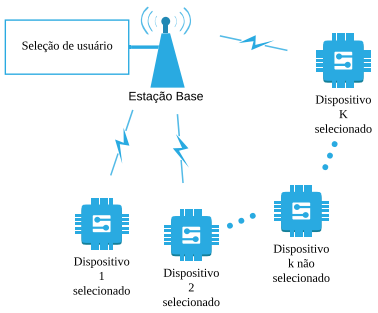

Fig. 1. Esquemático do funcionamento do sistema

As probabilidades de transição dos estados do *buffers* e dos canais de sistemas IdC cognitivo de comunicações sem fio são apresentados nas seções seguintes.

A. Estado dos Buffers

Cada um dos K dispositivos possuem um *buffer* de tamanho L, e em cada intervalo do sistema pacotes chegam e são transmitidos pelo dispositivo. Assim, o *buffer* de cada dispositivo pode apresentar filas de pacotes que variam de tamanho entre 0 e L pacotes. A probabilidade de chegar d pacotes em um determinado intervalo de tempo é de $p(d_i) = exp(-\lambda)\lambda^{d_i}/d_i!$, onde λ é a taxa de chegada em pacotes por intervalo de tempo. A quantidade de pacotes no *buffer* no intervalo de tempo posterior pode ser dada por:

$$l_{i+1,k} = min(d_{i,k} + l_{i,k} - t_{i,k}, L), \tag{1}$$

onde i é o intervalo de tempo, $l_{i,k}$ é a quantidade de pacotes para o dispositivo k, $d_{i,k}$ é o número de pacotes que chegam e $t_{i,k}$ a quantidade de pacotes que são transmitidos.

A probabilidade de transição dos estados dos *buffers* dos K dispositivos é o produtório das probabilidades individuais de cada dispositivo, ou seja:

$$p_l(l, l') = \prod_{k=1}^{K} p_{l,k}(l_i, l_{i+1}|a_{i,k}), \tag{2}$$

onde $a_{i,k}$ é o número de pacotes transmitidos pelo dispositivo k no intervalo i.

B. Estado dos Canais

Assumindo que a relação sinal-ruído (SNR) obedece a distribuição de Rayleigh [6], cuja função de densidade de probabilidade é $p(\rho) = 1/\bar{\rho}\ exp(-\rho/\bar{\rho})$, com o parâmetro $\rho > 0$ e $\bar{\rho} = E(\rho)$ sendo a SNR média. Seja o limiar da SNR expressado como $\rho_{snr} = \{\rho_1, \rho_2, \ldots, \rho_{C-1}\}$ e C o número de estados dos canais, pode-se obter a probabilidade de distribuição do estado do canal como:

$$p_C(c_n) = \int_{\rho_n}^{\rho_{n+1}} \mathrm{p}(\rho)\,\mathrm{d}\rho. \tag{3}$$

Assim, a probabilidade de transição do estado do canal é [7]:

$$p_C(c_n, c_{n+1}) = N(\rho_{n+1})T_f/p_C(c_n), \tag{4}$$

onde $n \in \{1, 2, \ldots, C-2\}$, e:

$$p_C(c_n, c_{n-1}) = N(\rho_n)T_f/p_C(c_n), \tag{5}$$

onde $n \in \{1, 2, \ldots, C-1\}$, T_f sendo a duração do intervalo e $N(\rho_n) = \sqrt{2\pi\rho_n/\bar{\rho}}f_d$ com f_d sendo o efeito doppler máximo. Analogamente à equação 2, a probabilidade de transição de estados dos M canais é:

$$p_C(c, c') = \prod_{m=1}^{M} p_{c,m}(c_i, c_{i+1}). \tag{6}$$

C. Potência

A transmissão de pacotes é realizada por diferentes modos de transmissão $j \in \{0, 1, ..., J\}$, onde os modos 0 e 1 representam nenhuma transmissão e transmissão BPSK respectivamente, e para $j \geq 2$, 2^j-QAM. Pode-se então estimar a potência mínima de transmissão P no estado de canal c_i com a modulação j a partir da taxa de erros de bit p_{BER} [8]:

$$p_{BER}(c_i, j) \leq 0,5erfc(\sqrt{\rho_i P(c_i, j)/WN_0}), \tag{7}$$

para $j = 1$, e para $j \geq 2$, tem-se:

$$p_{BER}(c_i, j) \leq 0,2exp(-1,6\rho_i P(c_i, j)/WN_0(2^j - 1)), \tag{8}$$

onde WN_0 é a potência de ruído.

III. O Sistema Como uma Cadeia de Markov

O sistema descrito na seção anterior, pode ser modelado como uma cadeia de Markov, uma vez que o estado seguinte depende somente do estado atual e da ação escolhida pelo agente [7]. Como o sistema permite que a transmissão ocorra de modo simultâneo, durante cada intervalo de tempo, o agente deverá escolher até no máximo M dispositivos para fazer a transmissão de seus respectivos pacotes, através dos M canais disponíveis, assumindo $M < K$, e usando modulações diferentes para cada canal. Assim, o conjunto de ações possíveis é uma junção de duas permutações sem repetições, e o número total de ações possíveis em um determinado estado é:

$$A = \frac{(J+1)!}{(J+1-M)!}\frac{K!}{(K-M)!}, \tag{9}$$

onde J é o número máximo dos modos de transmissão 2^j-QAM do sistema.

A. Probabilidade de Transição de Estados

Os estados do sistema são definidos por uma combinação dos estados dos *buffers* de cada dispositivo com os estados dos canais do sistema. Ambos podem ser definidos como uma permutação com repetição, uma dos K dispositivos tomados dos estados possíveis para cada dispositivo, e outra, dos M canais tomados dos estados possíveis dos canais. Assim, o número total de estados do sistema é:

$$S = (L+1)^K C^M, \tag{10}$$

onde L é o tamanho máximo do *buffer* e C é o número de estados dos canais. Assim, a probabilidade de transição de estados do sistema é:

$$p_S(S_i, S_{i+1}|a_i) = \prod_{k=1}^{K} p_{l_k}(l_i, l_{i+1}|a_i) \times \prod_{m=1}^{M} p_{c_m}(c_i, c_{i+1}). \tag{11}$$

B. Utilidade do Sistema

Em cada intervalo de tempo i, a vazão é definida como sendo o somatório do produto entre a taxa de codificação V e o modo de transmissão j para cada um dos K dispositivos, e o custo é definido como o produto da potência de transmissão consumida $P_{s_i}(s_i, a_i)$ com o somatório do valor da pressão do *buffer* $f_{i,k} = exp(\theta \times l_{i,k})$, com θ sendo o coeficiente de pressão.

A utilidade do sistema é diretamente proporcional ao número de pacotes transmitidos e inversamente proporcional à pressão dos *buffers* e do consumo de potência, sendo representada pela seguinte equação:

$$O(s_i, a_i) = \frac{\sum_{k=1}^{min(K,M)} V \times j_k}{\left(\sum_{k=1}^{K} f_{i,k}\right) P_{s_i}(s_i, a_i)}. \tag{12}$$

IV. Aprendizado por Reforço

O aprendizado por reforço é uma técnica que consiste em um agente tomando decisões em diversos estados de um ambiente e recebendo recompensas ou punições pelas suas ações [2]. Após uma série de testes de tentativa-erro, o agente busca aprender a melhor política, ou seja, a melhor sequência de ações a serem tomadas naquele ambiente de forma a obter valores de recompensas maiores.

Nesse artigo, o algoritmo de aprendizado por reforço *Q-learning* é utilizado, no qual é necessário obter as probabilidades de transição de estados e as recompensas de cada ação possível. Deve-se selecionar um fator de desconto que indica a relevância das recompensas futuras. Uma matriz **Q** de ação-valor é então gerada, com valores de utilidade esperados para cada ação realizada em cada estado, a matriz é atualizada a medida em que novas políticas são testadas. Para cada política π existe um valor agregado $V^\pi(s_i)$ e o objetivo do treinamento é fazer o agente aprender a determinar uma política que maximize $V^\pi(s_i)$ [2].

O seguinte algoritmo foi utilizado para treinar o agente com aprendizado por reforço:

Algoritmo Proposto 1: Algoritmo *Q-learning* baseado em Modelagem Markoviana do Sistema

- **Passo 1**: Calcule os valores da matriz **P** de probabilidade de transição de estados de acordo com a equação 11.
- **Passo 2**: Inicialize os valores da matriz **R** de recompensas usando a utilidade do sistema dada pela equação 12.
- **Passo 3**: Inicialize a matriz $\mathbf{Q} = \mathbf{0}$.
- **Passo 4**: Selecione um estado aleatório s_i e selecione uma ação aleatória a_i que seja possível de ser realizada no estado s_i.
- **Passo 5**: Simule para o novo estado s_{i+1} e associe com a recompensa associada a mudança de estado por conta da ação realizada.
- **Passo 6**: Atualize o valor de **Q**: $Q_{s_i,a_i} \leftarrow Q_{s_i,a_i} + \alpha(R_{s_i,a_i} + \gamma max(Q_{s_{i+1}}))$, onde α é a taxa de aprendizado do algoritmo e γ é a taxa de desconto.
- **Passo 7**: Selecione uma nova ação aleatória para o estado $s_i \leftarrow s_{i+1}$.
- **Passo 8**: Retorne ao passo 5 e repita por 100 iterações.
- **Passo 9**: Retorne ao passo 4 na próxima iteração, até o fim de todas as iterações.

Diferentes políticas podem ser criadas utilizando propostas diferentes na inicialização dos valores das recompensas, assim, pode-se priorizar outros aspectos de um mesmo sistema. Nesse artigo, a utilidade do sistema é utilizada como recompensa imediata das ações no treinamento do agente.

Com a matriz **Q** devidamente atualizada com o treinamento, faz-se então a comparação do desempenho da abordagem proposta com outros algoritmos de escalonamento da literatura através de simulações.

V. Algoritimos de Escalonamento

Algoritmos de escalonamento em sistemas de comunicação são utilizados para otimizar a utilização e compartilhamento dos recursos disponíveis [9]. Alguns algoritmos de escalonamento conhecidos na literatura foram escolhidos, os quais pode-se citar: *Earliest Deadline First* (EDF), *Logarithmic rule* (*LOG rule*), *Exponential rule* (*EXP rule*) e seleção aleatória (SA) [9].

O algoritmo EDF prioriza alocar recursos para dispositivos que possuam pacotes que chegaram a mais tempo.

A *EXP rule* estima o tempo de espera $w_k(i)$ para o dispositivo k num determinado intervalo de tempo e cria uma fila de espera baseado nesses tempos de espera, a seleção do dispositivo ocorre de acordo com uma maximização do argumento, dado pela seguinte equação:

$$k^*(i) \in \underset{1 \le i \le K}{\arg\max}\ b_k\, exp\left(\frac{a_k w_k(i)}{1 + \sqrt{(1/K)\sum_j w_j(i)}}\right) \times SE_k(i), \tag{13}$$

onde $SE_k(i)$ representa a eficiência espectral do dispositivo k. Similarmente à regra exponencial, o escalonador *LOG rule* é definida pela seguinte equação:

$$k^*(i) \in \underset{1\leq k\leq K}{\arg\max}\ b_k\, log\left(c + a_k w_k(i)\right) \times SE_k(i), \qquad (14)$$

os parâmetros a_k, b_k e c são constantes positivas arbitrárias, ver [10] sobre como esses parâmetros devem ser escolhidos.

O agente da seleção aleatória realiza o escalonamento de pacotes no sistema de transmissão sem fio de forma aleatória, independente do estado que o sistema se encontra. Na seção seguinte, os resultados obtidos nas simulações são apresentados.

VI. Simulação e Resultados

Para verificar e comparar a eficiência do algoritmo proposto utilizando aprendizado por reforço, foram realizadas simulações computacionais com outros algoritmos na literatura agindo sobre o mesmo sistema IdC cognitivo. O sistema assim como os cenários e algoritmos considerados foram implementados no Matlab.

Nas simulações do sistema IdC foram considerados 2 cenários diferentes. O primeiro cenário consiste em observar o comportamento dos algoritmos de alocação de recursos variando a taxa de chegada λ de 0,05 a 0,45 pacotes/ms. O valor do intervalo de tempo T_f de 1 milissegundo foi adotado por ser o intervalo de tempo de transmissão (TTI) em tecnologias 4G e e corresponde ao tamanho de um *slot* na numerologia 0 em 5G [6]. Nesse cenário, o sistema possui K = 4 dispositivos com um *buffer* de tamanho L = 3, M = 3 canais disponíveis, com C = 2 estados de canal, J = 4 modos de transmissão, ou seja, as modulações possíveis para a transmissão são BPSK, 4-QAM, 8-QAM, 16-QAM e nenhuma transmissão. A taxa de codificação considerada é V = 2. Para o segundo cenário, fixa-se o parâmtro da taxa de chegada em λ = 0,25 pacotes a cada milissegundo, altera-se o número de canais disponíveis para M = 2 e realiza-se a variação da quantidade de dispositivos, com K variando de 2 à 5 dispositivos. Os parâmetros a_k e b_k utilizados no *LOG rule* e *EXP rule* foram escolhidos aleatóriamente no intervalo entre 0 e 1, e $c = 1,1$. Os demais parâmetros usados na simulação podem ser encontrados na Tabela I.

TABELA I

Parâmetros usados na simulação.

Parâmetros	Valor
Coeficiente de pressão do *buffer*	$\theta = 0,5$
Limite de BER	$BER \leq 10^{-3}$
Potência de ruído	$10^{-3} W N_0/W$
Número total de intervalos de tempo	100
Parâmetro de Rayleigh	$\rho = 0,2$
Frequência máxima do efeito Doppler	$f_d = 50Hz$
Coeficiente de desconto	$\gamma = 0,5$
Taxa de aprendizado Q	$\alpha = 1/\sqrt{n_{ite}+2}$
Iterações do treinamento	$N = 100000$

As Figuras 2 à 6 apresentam gráficos comparativos relacionados ao desempenho dos 5 tipos de alocação: Seleção Aleatória, EDF, *LOG rule*, *EXP rule* e o algoritmo que utiliza Aprendizado por Reforço e Modelagem Markoviana.

Os resultados dos valores da utilidade do sistema do primeiro cenário para os algoritmos considerados nas simulações são apresentados na Figura 2. Verifica-se que com o aumento da taxa de chegada, os algoritmos apresentam maiores valores de utilidade, ocasionadas pelo aumento do número de pacotes transmitidos. O algoritmo proposto apresentou os maiores valores de utilidade dados pela equação 12 para valores de taxa de chegada acima de 0,15 pacotes por milissegundo e o *EXP Rule* apresentou os maiores valores de utilidade para valores de taxas de chegadas menores do que esse valor. Neste caso, um algoritmo que apresente maior valor de utilidade representa que o mesmo controla a transmissão de pacotes de forma mais eficiente, com uma melhor relação vazão e consumo de potência dos dispositivos.

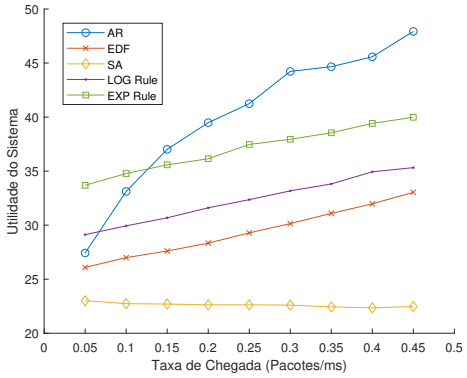

Fig. 2. Utilidade do sistema em relação à taxa de chegada para o cenário 1

Na Figura 3, visualizam-se os resultados da relação entre a taxa de pacotes perdidos pela taxa de pacotes transmitidos dos algoritmos considerados na simulação. Os algoritmos EDF e AR apresentaram os menores valores de taxas de perda de pacotes ao longo da variação da taxa de chegada.

Na Figura 4, pode-se observar os resultados normalizados para a quantidade de pacotes de dados transmitidos. Observa-se que os valores apresentados pelos algoritmos apresentam um crescimento quando se aumenta a taxa de chegada λ. Destaca-se que o algoritmo proposto apresentou os maiores valores de quantidade de pacotes transmitidos para $\lambda > 0.15$ pacotes/ms.

No segundo cenário, com o sistema mais limitado com menos canais disponíveis, a utilidade do sistema, mostrado na Figura 5, decresce com o aumento de dispositivos devido à dificuldade em se controlar a pressão dos *buffers*, que aumenta o custo do sistema. Os algoritmos AR e EDF obtiveram os maiores valores.

As taxas de pacotes perdidos por pacotes transmitidos são mostrados na Figura 6, que crescem com o aumento do número de dispositivos. Os algoritmos que proporcionaram menos perdas de pacotes para o sistema foram o AR e o EDF.

Com a realização das simulações, pode-se observar que o desempenho do algoritmo AR Markoviano em termos de perda de pacotes e ocupação do *buffer* é comparável aos melhores resultados obtidos pelos algoritmos considerados, principalmente aqueles obtidos pelo algoritmo EDF. Entretanto, o

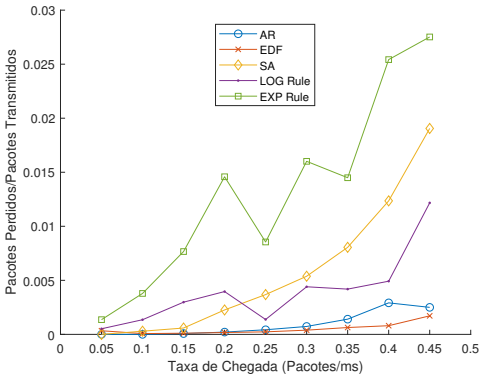

Fig. 3. Taxa de pacotes perdidos por pacotes transmitidos em relação à taxa de chegada para o cenário 1

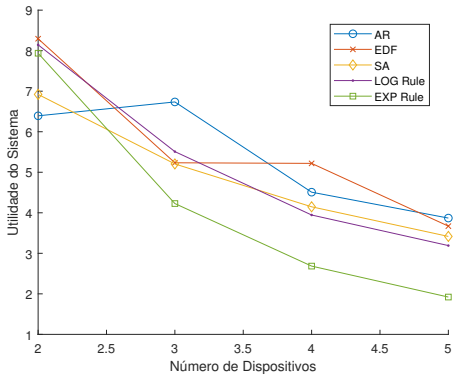

Fig. 5. Utilidade do sistema em relação ao número de dispositivos para o cenário 2

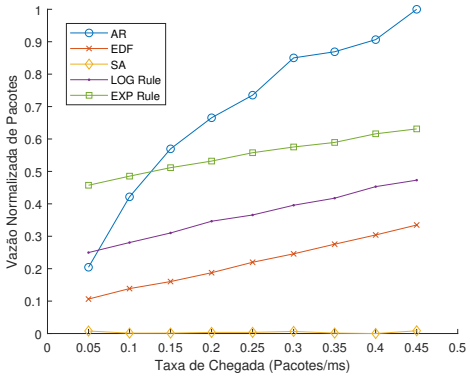

Fig. 4. Quantidade de pacotes transmitidos em relação à taxa de chegada para o cenário 1

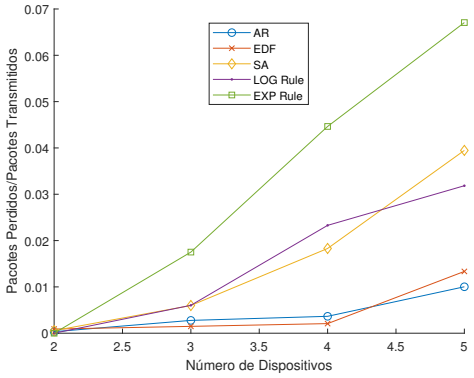

Fig. 6. Taxa de pacotes perdidos por pacotes transmitidos em relação ao número de dispositivos para o cenário 2

algoritmo AR baseado em Cadeia de Markov se destaca por conseguir valores altos de utilidade e vazão, mantendo uma taxa pequena de perda de pacotes comparado aos outros algoritmos tanto no cenário 1 quanto no cenário 2.

VII. Conclusões

Nesse artigo, apresenta-se uma proposta de algoritmo utilizando aprendizado por reforço baseada em Cadeia de Markov para realizar o escalonamento na transmissão de pacotes em um sistema de comunicação IdC cognitivo sem fio com múltiplos dispositivos. Para tal, adota-se uma cadeia de Markov para modelar os estados do sistema de comunicação e suas transições, fornecendo os parâmetros necessários para determinar ações de alocação de recursos de acordo com o Algoritmo Proposto 1.

Os resultados apresentados referentes às simulações computacionais de um sistema IdC cognitivo mostram que o aprendizado por reforço tem um desempenho em geral superior em relação aos outros algoritmos de escalonamento considerados. O algoritmo AR proposto se destaca por apresentar valores de utilidade de sistema maiores nos dois cenários simulados, onde o agente é treinado para priorizar a vazão de pacotes levando em consideração também o consumo de potência envolvido na transmissão e uma variável relacionada à pressão de *buffer*.

Referências

[1] S. Jang, H. Yoon, N. Park, J. Yun, and Y. Son, "Research trends on deep reinforcement learning," *Electronics and Telecommunications Trends*, vol. 34, no. 4, pp. 1–14, 2019.

[2] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L.-C. Wang, "Deep reinforcement learning for mobile 5g and beyond: Fundamentals, applications, and challenges," *IEEE Vehicular Technology Magazine*, vol. 14, pp. 44–52, 06 2019.

[3] C. Zhu, X. Cheng, H. Ye, J. Yang, L. Xu, and K. Chao, "5g wireless networks meet big data challenges, trends, and applications," pp. 1513–1516, 08 2019.

[4] Q. Wei, D. Liu, and G. Shi, "A novel dual iterative q-learning method for optimal battery management in smart residential environments," *Industrial Electronics, IEEE Transactions on*, vol. 62, pp. 2509–2518, 04 2015.

[5] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Transactions on Wireless Communications*, vol. PP, pp. 1–1, 11 2018.

[6] A. Zaidi, F. Athley, J. Medbo, U. Gustavsson, G. Durisi, and X. Chen, *5G Physical Layer: principles, models and technology components*. Academic Press, 2018.

[7] J. Grewal, M. Krzywinski, and N. Altman, "Markov models—markov chains," *Nature Methods*, vol. 16, pp. 663–664, Aug. 2019.

[8] S. Haykin, *Digital Communication Systems*. Wiley, 2013.

[9] B. Sadiq, R. Madan, and A. Sampath, "Downlink scheduling for multiclass traffic in lte," *EURASIP J. Wirel. Commun. Netw.*, vol. 2009, Mar. 2009.

[10] S. Shakkottai and A. Stolyar, "Scheduling algorithms for a mixture of real-time and non-real-time data in hdr," *Teletraffic Science and Engineering*, vol. 4, 09 2001.