Soluções baseadas em aprendizado por reforço profundo para implantar VANTs como gateways LoRaWAN com foco na Qualidade de Serviço IoT

Rogério S. Silva, Renan R. Oliveira, Lucas T. S. Carvalho, Leandro A. Freitas,
Xavier P. Sebastião, Cleyber B. Reis, Antonio Oliveira-Jr, e Kleber V. Cardoso

***Resumo*— O uso de Veículos Aéreos Não Tripulados (VANTs) é uma estratégia eficaz para densificar redes de acesso sem fio, melhorando o desempenho de aplicações *Internet of Things* (IoT) sob demanda. Este artigo associa o fatiamento de redes 3^{rd} *Generation Partnership Project* (3GPP) com os parâmetros de QoS da tecnologia LoRa não-3GPP, para garantir desempenho adequado aos dispositivos IoT, ou seja, os requisitos de QoS dos *slices* não-3GPP. Formulamos um problema de otimização Programação Linear Inteira Mista (*Mixed Integer Linear Programming* – MILP) para minimizar o número de VANTs e determinar suas posições, o qual se mostrou NP-difícil. Como alternativa, modelamos o problema como um Processo de Decisão de Markov (*Markov Decision Process* – MDP) e propusemos soluções baseadas em *Deep Q-Network* (DQN) e *Advantage Actor-Critic* (A2C) para posicionar os VANTs. Essas soluções foram integradas a simulações *on-line* com o *Network Simulator 3* (ns-3), resultando em melhorias significativas nos níveis de QoS em comparação com outras soluções do estado da arte.**

***Palavras-Chave*— DRL, DQN, A2C, VANTs, LoRaWAN, ns-3.**

***Abstract*— Using *Unmanned Aerial Vehicle*s (UAVs) is an effective strategy for densifying wireless access networks, improving the performance of IoT applications on demand. This article combines 3GPP network slicing with QoS parameters of LoRa non-3GPP technology to ensure adequate performance for IoT devices, i.e. QoS requirements of non-3GPP *slices*. We formulated an optimization problem MILP to minimize the number of UAVs and their respective positions, which proved to be NP-hard. As an alternative, we modeled the problem as an MDP and proposed solutions based on DQN and A2C to position the UAVs. These solutions have been integrated into *online* simulations with ns-3, resulting in significant improvements in QoS levels compared to other state-of-the-art solutions.**

***Keywords*— DRL, DQN, A2C, UAVs, LoRaWAN, ns-3.**

I. Introdução

A IoT tem revolucionado as comunicações sem-fio e as tecnologias de redes móveis. As redes móveis de 5^a e 6^a gerações estão sendo desenvolvidas na perspectiva de, entre outras, suportar Comunicações Massivas do Tipo Máquina (*Massive Machine Type Communications* – mMTC), oportunizando cenários de grande demanda por aplicações IoT. Espera-se que nos próximos anos o número de dispositivos conectados alcance algumas dezenas de bilhões [1]. Esse crescimento no número de dispositivos conectados afetará significativamente as comunicações, impactando as garantias de QoS.

Várias tecnologias emergentes propõem lidar com o crescimento da IoT e, entre essas, as *Low Power Wide Area Network*s (LPWANs) destacam-se no atendimento às demandas da IoT, com capacidade de acesso a longas distâncias. *Long Range Wide Area Network* (LoRaWAN), Sigfox, NB-IoT, e LTE-M, são as principais alternativas entre as LPWANs. Nesse contexto, LoRaWAN se destaca por ser uma tecnologia baseada em um padrão aberto e prover comunicações de longo alcance, com altas taxas de dados, segurança e baixo consumo energético [2]. O protocolo LoRaWAN é implementado na camada *Media Access Control* (MAC) e construído sobre a camada física LoRa. A pilha de protocolos LoRaWAN possibilita que os *LoRa End Device*s (LoRa-EDs) estabeleçam comunicação de longo alcance, e provê um mecanismo que permite a adaptação dinâmica dos parâmetros de *Spreading Factor* (SF) e de *Transmission Power* (TP). O *Adaptive Data Rate* (ADR) possibilita que os LoRa-EDs possam otimizar o consumo energético e as taxas de transmissão [3].

Infraestruturas de comunicação tradicionais muitas vezes enfrentam falhas devido a interrupções na rede ou em ambientes de alta densidade. Os VANTs têm sido adotados como infraestrutura complementar para as redes, como alternativa de baixo custo e alta flexibilidade, para prover serviços de comunicação [4], [5]. VANTs têm o potencial de melhorar significativamente a QoS devido à sua capacidade de implantação flexível, baixa latência e alocação eficiente de recursos [5].

Implantar VANTs para prover serviços de comunicação e que atendam aos requisitos de aplicações não-3GPP se enquadra como um problema de otimização. Desta forma, este trabalho, inicialmente, propõe uma abordagem baseada em MILP com objetivo de minimizar o número de VANTs necessários para atender a demanda não-3GPP, definir as posições de implantação desses VANTs e ainda atender às restrições impostas para alcançar a QoS desejada. Propôs-se então uma abordagem de Aprendizado por Reforço Profundo (*Deep Reinforcement Learning* – DRL), modelada como um MDP e implementada especificamente como *Deep Q-Learning* (DQL) combinado com o método de políticas de gradiente

Rogério S. Silva, Renan R. Oliveira, Leandro A. Freitas e Lucas T. S. Carvalho, Instituto Federal de Goiás (IFG), e-mail: {rogerio.sousa, renan.rodrigues, leandro.freitas}@ifg.edu.br, lucastsc@gmail.com; Antonio Oliveira-Jr, Kleber V. Cardoso, Cleyber B. Reis e Xavier P. Sebastião, Universidade Federal de Goiás (UFG), e-mail:{antonio, kleber, cleyber.bezerra, xaviersebastiao}@inf.ufg.br; Antonio Oliveira-Jr, Fraunhofer Portugal AICOS, Portugal. Este trabalho foi parcialmente financiado por CAPES, MCTIC/CGI.br/Fundação de Amparo à Pesquisa de São Paulo (FAPESP) por meio do Projeto Smart 5G Core And MULtiRAn Integration (SAMURAI) 2, pelo CNPq por meio do Projeto Universal sob Bolsa 405111/2021-5, e pela RNP/MCTIC, Outorga nº 01245.010604/2020-14, no âmbito do projeto Sistemas de Comunicações Móveis 6G.

A2C para lidar com a complexidade desse ambiente.

Este trabalho está organizado como segue. Na Seção II apresentamos os trabalhos relacionados. A Seção III é dedicada à modelagem do sistema. Formulamos o problema como um MILP e o aprimoramos para um MDP na Seção IV. Os resultados são apresentados na Seção V e finalmente concluimos e apresentamos as considerações finais na Seção VI.

II. Trabalhos Relacionados

Este trabalho considera a integração de VANTs, redes IoT não-3GPP LoRaWAN, e *Network Slicing* (NS). A Tabela I apresenta uma síntese dos trabalhos relacionados.

TABELA I: Características dos Trabalhos Relacionados.

Trabalhos relacionados	**5G**	**N3**	**NS**	**NT**	**VA**	**PL**	**DQ**	**AC**
Dawaliby, et al., 2019 [6]		✓	✓	✓		✓		
Dawaliby, et al., 2021 [7]		✓	✓	✓		✓		
Telache, et al., 2022 [8]			✓	✓			✓	
Mardi, et al., 2022 [9]		✓	✓	✓		✓		
Marchese, et al., 2020[10]		✓		✓	✓			
Mahmood, et al., 2022[11]	✓			✓	✓			
Almeida, et. al., 2022 [12]				✓	✓		✓	
Silva R. S., et al., 2023 [13]	✓	✓	✓	✓	✓	✓		
Nossa proposta	✓	✓	✓	✓	✓	✓	✓	✓

5G–3GPP **N3**–Não-3GPP **NS**–*Slicing* **NT**–*Net.Tunning* **VA**–VANTs **PL**–Otimização **DQ**–DQN **AC**–A2C

Trabalhos recentes têm proposto métodos para fatiamento de redes LoRaWAN, por meio de algoritmos que ajustam parâmetros de comunicação (*network tunning*). Dawaliby et al. [6], [7], apresentam um fatiamento dinâmico baseado em estimativas de máxima similaridade com ênfase em evitar a escassez de recursos e priorizar *slices* de acordo com seus requisitos. Tellache et al. [8], propõe o uso de DQN para alocação de recursos para *slices* em redes LoRaWAN densas. Mardi et al. [9], aplicam a teoria dos jogos para gerenciar de forma mais eficiente os nós LoRa. Esses trabalhos consideram estratégias para fatiamento de redes LoRaWAN, porém atuam nos ajustes dos parâmetros do ponto de vista dos dispositivos LoRa e não do reposicionamento dos *LoRa Gateway*s (LoRa-GWs). A presente proposta, além de acomodar os dispositivos em fatias de rede, consideram também outros parâmetros da rede, e.g., atraso e interferência como entrada para modelos de otimização para reposicionar os LoRa-GWs implantados em VANTs e melhorar a QoS global do sistema.

Os trabalhos [10], [11], [12] adotam VANTs como estações de base móveis, para otimizar e expandir a infraestrutura de rede. Marchese et al. [10], empregam os VANTs como gateways LoRaWAN integrados a satélites para estender a cobertura de rede. Mahmood et. al. [11], apresentam uma otimização por enxame de partículas (PSO) para maximizar as taxas de dados e otimizar a implantação de VANTs. Almeida et al. [12], utilizam *Deep Reinforcement Learning* (DRL) para maximizar a utilidade da rede de acordo com a demanda dos usuários. A presente proposta, considera a implantação de gateways LoRaWAN em VANTs, aproveitando sua flexibilidade de implantação e de posicionamento. Todavia, o principal foco está nas estratégias de otimização e de aprendizado por reforço, de modo a reduzir custos inerentes a minimização do número de VANTs, por associar os dispositivos LoRa em *slices*, e por reconfigurar seus parâmetros de SF e TP com foco na melhoria da QoS e redução de interferências.

Em nosso trabalho anterior [13], propusemos um MILP com objetivo de minimizar o número de VANTs e os posicionar para alcançar melhores índices de QoS. No presente trabalho, expandimos o espaço de busca, incorporamos novas restrições, empregamos técnicas de aprendizagem por reforço profundo para melhorar a robustez do modelo e propomos duas abordagens de solução baseadas em DQN e A2C, respectivamente.

III. Modelagem do Sistema

Seja uma rede definida por um conjunto de LoRa-EDs sem mobilidade e aleatoriamente distribuídos e um conjunto de LoRa-GWs instalados em VANTs. Considera-se também que cada LoRa-GW se comunica com *Base Station*s (BSs) 5^{th} *Generation Networks* (5G), operando em banda de frequência sub6-GHz, configuradas com parâmetros que atendam aos requisitos do LoRa-GWs, não impondo nenhuma restrição aos mesmos. Dessa forma, objetivamos minimizar o número de LoRa-GWs necessários para atender aos requisitos de QoS estabelecidos e encontrar as melhores posições para implantar os VANTs. Para isso, minimizamos o número de posições em um espaço discreto alocado para implantá-los.

Seja $\mathcal{K} = \{k_1, ..k_k, .., k_{|\mathcal{K}|}\}$ o conjunto de LoRa-EDs conectados aos LoRa-GWs e pertencentes ao conjunto de *slices* $\mathcal{L} = \{l_1, ..l_l .., l_{|\mathcal{L}|}\}$, os *slices* são definidos baseados nos requisitos de QoS das respectivas aplicações IoT, e $\mathcal{C} = \{c_1, ..c_c .., c_{|\mathcal{C}|}\}$ o conjunto de configurações com as possiveis combinações entre SF e TP, $\mathcal{C} \subseteq (\mathcal{SF} \times \mathcal{TP})$. O espaço onde os LoRa-GWs poderão ser implantados foi discretizado no conjunto $\mathcal{P} = \{p_1 = (x_1, y_1, z_1),\ p_2 = (x_2, y_2, z_2),\ \ldots,\ p_{|\mathcal{P}|} = (x_{|\mathcal{P}|}, y_{|\mathcal{P}|}, z_{|\mathcal{P}|})\}$ de pontos equidistantes e uniformemente distribuídos com distancia d em três eixos perpendiculares entre si. Assim, definimos a função objetivo como,

$$\min \quad \sum_{p \in \mathcal{P}} \left\lceil \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \frac{x^p_{k,c}}{|\mathcal{K}|} \right\rceil, \tag{1}$$

onde $x^p_{k,c} \in \{0, 1\}$ é a variável de decisão que indica se um VANT refere-se a um LoRa-GW posicionado no ponto $p \in \mathcal{P}$ para atender o LoRa-ED $k \in \mathcal{K}$ com a configuração $c \in \mathcal{C}$. O modelo está sujeito às restrições definidas em [13] a seguir.

Com relação ao posicionamento dos LoRa-GWs:

$$\left\lceil \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \frac{x^p_{k,c}}{|\mathcal{K}|} \right\rceil \leq 1, \quad \forall p \in \mathcal{P}, \tag{2}$$

assegura que cada posição $p \in \mathcal{P}$ possa ser ocupada por somente um VANT LoRa-GW. Com relação à associação de LoRa-EDs e LoRa-GWs:

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x^p_{k,c} = 1, \quad \forall k \in \mathcal{K}, \tag{3}$$

garante que cada LoRa-ED $k \in \mathcal{K}$ se conecte apenas a um LoRa-GW na posição $p \in \mathcal{P}$ com a configuração específica $c \in \mathcal{C}$. Com relação ao tráfego de *uplink*,

$$\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \mathcal{S}_{k,l} \cdot x^p_{k,c} \cdot R_k \leq R^{max}_l, \forall l \in \mathcal{L}, \forall p \in \mathcal{P}, \tag{4}$$

onde, $\mathcal{S}(k, l) \in \{0, 1\}$ é uma função de mapeamento que retorna 1 quando o LoRa-ED $k \in \mathcal{K}$ está associado ao *slice* $l \in \mathcal{L}$ e 0 caso contrário. $\mathcal{R}_k$ representa a soma do tráfego

de *uplink* para o LoRa-EDs em $l \in \mathcal{L}$ não deve exceder a capacidade de tráfego $\mathcal{R}_l^{max}$ de l.

Com relação ao alcance dos LoRa-EDs aos LoRa-GWs,

$$x_{k,c}^p \leq I_{k,c}^p, \quad \forall p \in \mathcal{P}, \forall k \in \mathcal{K}, \forall c \in \mathcal{C}, \tag{5}$$

onde $I_{k,c}^p \in \{0,1\}$ indica se k alcança algum LoRa-GW em $p \in \mathcal{P}$, com a configuração $c \in \mathcal{C}$.

A. QoS

Assume-se que os *LoRa Network Server*s (LoRa-NSs) são cientes dos requisitos de QoS de cada LoRa-ED em termos de atraso e taxa de dados e que os LoRa-NSs são responsáveis por definir as estratégias de alocação de recursos nos LoRa-GWs e por configurar os LoRa-EDs com os parâmetros de SF e TP. Cada dispositivo $k \in \mathcal{K}$ adota uma configuração específica de SF para transmissão de informações. A taxa de dados r_k, o atraso d_k e a QoS são calculados segundo [13]. A taxa de dados é obtida por $r_k = SF_c \cdot \frac{b_l}{2^{SF_c}} \cdot CR$ bits/s, o atraso é definido como $d_k = \frac{\sigma}{r_k}$ s, e σ é o tamanho do pacote em bits. assim, o custo de QoS é então obtido por $QoS_k = \overline{r_k} + (1 - \overline{d_k})$, onde $\overline{r_k}$ e $\overline{d_k}$ são normalizados dividindo-os pelos maiores valores possíveis de taxa de dados e atraso de um enlace LoRa.

Assim, definimos a restrição relativa à QoS:

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x_{k,c}^p \cdot S_{k,l} \cdot QoS_k \geq \rho_l^{QoS}, \quad \forall l \in \mathcal{L}, \forall k \in \mathcal{K}, \tag{6}$$

onde QoS_k é a QoS de cada dispositivo $k \in \mathcal{K}$ em um LoRa-GW, e ρ_l^{QoS} é a constante que representa o limite inferior permitido para o QoS.

IV. Formulação do Problema

Na seção III apresentamos o modelo de sistema para o problema conjunto de minimizar o número de VANTs, otimizar o posicionamento e definir as configurações dos LoRa-EDs. Todavia, o problema demostrou ser NP-difícil, ou seja não há soluções em tempo-polinomial para solucioná-lo. Para demonstrar que o problema é NP-difícil, reduzimos o problema de otimização da cobertura de conjuntos, *Set Cover Optimization* (SCO), que é conhecido por ser NP-difícil, para o problema proposto, ou seja, mapeamos cada elemento $u \in U$ para um LoRa-ED $k \in \mathcal{K}$ e cada subconjunto S_i para uma possível posição $p \in \mathcal{P}$ de um LoRa-GW. A cobertura de um elemento u por um subconjunto S_i corresponde ao atendimento de um LoRa-ED k por um LoRa-GW em uma posição p. Assim, encontrar k posições de LoRa-GWs que minimizam a função objetivo (1) e satisfazem as restrições (2), (3), (4), (5), e (6) é equivalente a resolver o SCO. Portanto, segue que o problema proposto também é NP-difícil, como pretendíamos demonstrar. Assim, apresentaremos a formulação do problema como um MDP e os algoritmos DQN e A2C que buscam solucioná-lo. Especificamente, modelamos o problema de posicionamento da seguinte forma.

A. MDP para o posicionamento de VANTs

O modelo MDP é proposto como uma estrutura para tomada de decisão, consistindo em cinco componentes principais: $<\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma>$, onde $\mathcal{S}$ representa o conjunto de estados, $\mathcal{A}$ denota o espaço de ação, $\mathcal{T}$ é a função de transição, $\mathcal{R}$ representa a função de recompensa e γ é o fator de desconto.

O espaço de estados é definido por $\mathcal{S} = \{s_1, s_2, ..., s_n\}$, onde cada s_i é uma possível combinação de posições dos VANTs à serem implantados em $\mathcal{P}$. Assim $|\mathcal{S}|$ é obtido por:

$$C_{(|\mathcal{P}|,\delta)} = \frac{|\mathcal{P}|!}{(\delta! \cdot (|\mathcal{P}| - \delta)!)}, \tag{7}$$

onde δ é o número de VANTs a serem implantados. O conjunto das possíveis ações foi definido como $\mathcal{M} = \{m_{norte}, m_{sul}, m_{leste}, m_{oeste}, m_{\otimes}\}$, onde m_{norte} indica mover o VANT para a direção cardeal norte, e assim para as outras direções cardeais e $m_{\otimes}$ indica que o VANT permanecerá parado na mesma posição. Desta forma, o espaço de ações é definido pelo total de possíveis arranjos com repetições $\mathcal{A} = \{a_1, a_2, ..., a_{|\mathcal{A}|}\}$, e $|\mathcal{A}| = |\mathcal{M}|^{\delta}$, e.g., se $\delta = 2$, então $\mathcal{A}$ conterá as ações $\{(m_{norte}, m_{norte}), (m_{norte}, m_{sul}), ..., (m_{\otimes}, m_{\otimes})\}$, totalizando $5^2 = 25$ ações. Assim, cada ação será uma tupla, indicando o movimento de cada VANT a ser posicionado.

A função de transição representa a probabilidade de mudar de um estado s para o estado s' por meio da ação a e é definido como $\mathcal{T}(s'|s,a) = \wp(s'|s,a)$, onde $\wp(s'|s,a)$ produz uma distribuição de probabilidade sobre os estados para os quais o sistema pode fazer a transição ao tomar a ação a a partir do estado s. A função de recompensa $\mathcal{R}(s'|s,a)$ retorna a recompensa imediata de quando se toma a ação a a partir do estado s. A recompensa é obtida a partir de:

$$\mu QoS_s = \frac{\sum QoS_{k,s}}{|K|}, \forall k \in K \tag{8}$$

$$\mathcal{R}(s'|s,a) = \begin{cases} \mu QoS > \rho_l^{QoS} & : \quad +\mu QoS \\ \mu QoS < \rho_l^{QoS} & : \quad -\mu QoS \\ \mu QoS = \rho_l^{QoS} & : \quad 0 \end{cases}$$

$$\gamma = \begin{cases} -1 & : \text{ ao sair da área} \\ -2 & : \text{ em caso de colisões,} \end{cases}$$

onde μQoS_s é a média de QoS obtida em um estado s, e ρ_l^{QoS} é a constante que representa o limiar desejado de QoS. Ressalta-se que os valores -1 e -2 são penalizações por ações indesejadas, como saída da área delimitada, e colisões entre VANTs. A Rede Neural Profunda (DNN) usa a função de recompensa para treinamento enquanto satisfaz a função objetivo da Equação 1. O fator de desconto γ determina o quanto as futuras recompensas devem ser descontadas após $\mathcal{T}(s'|s,a)$. Dessa forma, se γ tiver valores próximos de zero, as recompensas imediatas serão priorizadas, caso contrário, o foco muda para recompensas de longo prazo.

B. Algoritmo DQN

DQNs têm sido utilizadas para lidar com problemas de posicionamento de VANTs [14]. O algoritmo DQN utiliza DNN em conjunto com *Q-Learning* (QL) para solucionar problemas de aprendizado por reforço com espaços de estados complexos e de alta dimensionalidade [8]. Nesse contexto, propusemos uma arquitetura para a rede neural com 4 camadas, sendo a camada de entrada (l_1) representada por $\mathcal{S}$, as camadas ocultas l_2 e l_3 com 150 e 100 neurônios respectivamente, e a camada de saída l_4 representada por $\mathcal{A}$. Nossa abordagem é apresentada no Algoritmo DQN-LoVQI 1.

Utilizamos o ns-3 como ambiente de execução para o agente DQN, e realizamos a integração por meio do *framework* ns3-gym [15]. A linha 2 representa a definição do ambiente ns-3

Algorithm 1 Algoritmo DQN-LoVQI

1: **Inicialização:**
2: map ← NS3Environment()
3: **Modelo:**
4: $l1, l2, l3, l4$ ← Camadas($\mathcal{S}$, z, ac, W, b, Linear(), ReLU(), $\mathcal{A}$)
5: F.Perda, T.Ap, Opt ← MSELoss(), $(1e-3)$, Adam()
6: **for** episódio ← 0até|episódios| **do**
7: $s, \epsilon \leftarrow$ NS3Random(), ϵ_0
8: **for** passo ← 0 até |passos| **do**
9: $a \leftarrow \epsilon$-greedy()
10: $s', \mathcal{R} \leftarrow$ NS3DoAction(argmax(s), a)
11: $Y \leftarrow \mathcal{R} + (\gamma \times$ maxQ)
12: $Y_{pred} \leftarrow Q(s, a)$
13: loss ← MSELoss(Y_{pred}, Y)
14: $s \leftarrow s'$
15: **end for**
16: **end for**

integrado ao agente DQN. Nas linhas 4 e 5, configuramos os parâmetros básicos para o DQN e codificamos a rede neural como uma série de transformações lineares z seguidas por funções de ativação $ac(ReLU)$, pelos pesos W e pelos vieses b. As linhas 7 e 10 mostram, respectivamente, a inicialização do estado e a execução de uma ação no ns-3. Os laços das linhas 6 e 8 dirigem o processo temporal do algoritmo.

C. Algoritmo A2C

O algoritmo A2C usa uma técnica de DRL que combina elementos de aprendizado baseado em política (Ator) e aprendizado baseado em valor (Crítico). No A2C, um ator gera ações com base em uma política estocástica, enquanto um crítico avalia o valor das ações tomadas pelo ator em relação ao estado atual. Em comparação com o DQN, o A2C apresenta vantagens como uma convergência mais rápida, menor variância nas estimativas de valor e uma melhor exploração do espaço de ação [14]. Nesse ínterim, propomos o Algoritmo A2C-LoVQI com objetivo de aprimorar os resultados obtidos, minimizar a variância nas estimativas de $Q(s|a)$ e melhorar a exploração de $\mathcal{S}$.

Algorithm 2 Algoritmo A2C-LoVQI

1: **Inicialização:**
2: map ← NS3Environment()
3: **Modelo:**
4: $\text{DNN}_{\text{Actor}}$ ← Camadas($\mathcal{S}$, z, ac, W, b, $tanh()$, $\mathcal{A}$)
5: DNN_{Critic} ← Camadas($\mathcal{S}$, z, ac, W, b, $ReLu()$, $\mathcal{A}$)
6: F.Perda, T.Ap, Opt ← MSELoss(), $(1e-3)$, Adam()
7: **for** episódio ← 0 até episódios **do**
8: estado ← $NS3Random()$
9: **for** passo ← 0 até passos **do**
10: ação ← $torchCategorical()$
11: $s', \mathcal{R} \leftarrow NS3DoAction()$
12: advantage ← $\mathcal{R} + (1 - done) * \gamma * vCritic(s') - vCritic(s)$
13: DNN_{Critic} ← argMin; MSEError(advantage)
14: DNN_{Actor} ← argMax; log $\wp(s'|s, a)$
15: **end for**
16: **end for**

No Algoritmo 2, as linhas 2, 6, 7, 8, 9 e 11 têm o mesmo comportamento relatado na Seção IV-B. As linhas 4 e 5 configuram as redes neurais do Ator e do Crítico, nas linhas de 12–14 são computadas a vantagem e as atualizações das DNN Ator e Crítico. Finalmente os laços das linhas 7 e 9 conduzem o algoritmo A2C.

V. Avaliações

A. Ambiente de simulação

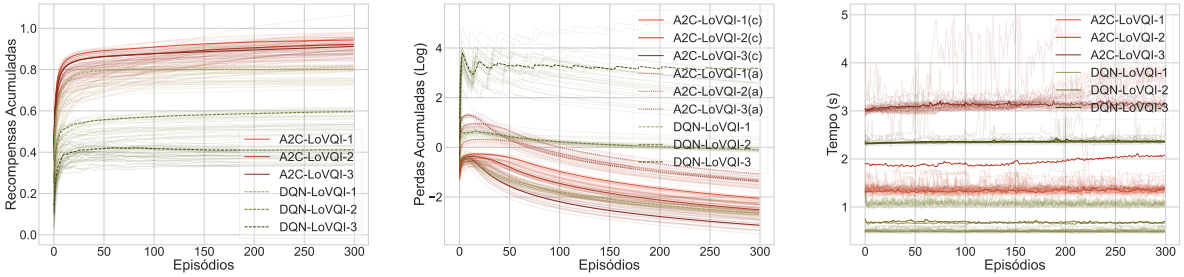
Consideramos um modelo de simulação através do ns-3 com a utilização dos módulos Sliced-LoRaWAN[1] e ns3gym [15]. Modelamos uma rede dinâmica e integrada aos agentes de DRL de modo que o agente fosse capaz de instanciar o ns-3 como um *environment* durante sua execução. Assim, o agente conecta-se ao ns-3, informa o estado, a ação a ser executada, e os parâmetros de configuração da rede e aguarda como retorno os resultados da simulação.

Para cada experimento, distribuimos conjuntos de 10, 30 e 50 LoRa-EDs, seguindo um padrão de tráfego realista adaptado de [16], em um ambiente urbano com dimensão 10 Km^2. Definimos o espaço para implantação de VANTs com $|\mathcal{P}| = n^2, n = 10$, uniformemente espaçados e com altitude de 45m em conformidade com a Agência Nacional de Aviação Civil (ANAC) [17]. Sendo $\mathcal{X}$ o número de LoRa-GW a serem implantados, resultante do MILP, limitamos os casos nos quais $\mathcal{X} \in \{1, 2, 3\}$, e definimos três tipos de *slices* com suas respectivas R_l^{max}. Conduzimos dez experimentos, integrando agentes e ns-3, com sementes aleatórias (MRG32k3a). Os resultados apresentam os valores com intervalo de confiança de 95%. Avaliamos duas estratégias distintas baseadas nos Algoritmos 1 e 2 e as comparamos com os resultados obtidos da implementação do modelo MILP apresentado em [13].

B. Resultados

Inicialmente, comparamos o A2C-LoVQI com o DQN-LoVQI. Observa-se na Fig. 1(a), as recompensas acumuladas, nesse contexto, os agentes tendem a convergir para um nível de desempenho estável após aproximadamente 40 episódios, indicado pela estabilização das linhas. Os agentes A2C-LoVQI demonstram um desempenho superior em comparação com os DQN-LoVQI. A Fig. 1(b) apresenta as perdas acumuladas em relação aos episódios em escala logarítmica, no geral, os agentes apresentam tendência de queda nas perdas ao longo dos episódios, o que sugere que estão melhorando sua capacidade de tomar decisões ótimas à medida que ganham experiência. Os agentes DQN apresentam perdas maiores e maior instabilidade das curvas, demonstrando desempenho inferior ao A2C, quando comparados segundo o mesmo número de VANTs. A Fig. 1(c) reflete os tempos de execução dos episódios para os dois métodos e suas variações em $\mathcal{X}$. Percebe-se que em média, os métodos A2C, em variações de tons vermelhos, demandam maior tempo de processamento que os respectivos DQN, em variações de verde, isso se dá pela natureza dos algoritmos. O A2C-LoVQI utiliza duas redes neurais separadas, além de calcular o gradiente para estas redes. Apesar de demandar maior tempo de processamento, o método A2C-LoVQI apresenta melhores resultados gerais, alcançando maiores níveis de QoS que o DQN-LoVQI.

Avaliamos também o desempenho do A2C-LoVQI em comparação com os resultados do DQN-LoVQI e do *baseline* implementado como MILP. A Fig. 2 apresenta os resultados da execução do simulador para os métodos *baselines* MILP em comparação com os resultados das simulações executadas sobre os resultados dos algoritmos A2C-LoVQI e DQN-LoVQI, para 1, 2 e 3 VANTs. Cabe ressaltar que o *baseline* para um VANT (Bas_1) não obteve resultados para 30 e 50 LoRa-EDs, assim como o Bas_2, para 50 LoRa-EDs, o que evidencia as limitações de execução do MILP. Considerando o Bas_3 como o resultado que considera as posições ótimas para três VANTs, pode-se observar que os resultados do algoritmo A2C-LoVQI alcançaram patamares muito próximos, mesmo



(a) Recompensas acumuladas. (b) Perdas acumuladas. (c) Tempo decorrido.

Fig. 1: Métricas para A2C-LoVQI e DQN-LoVQI por quantidade de VANTs.

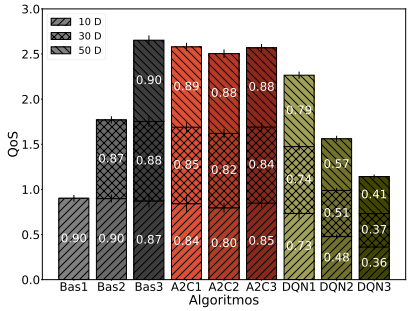


Fig. 2: Qos médio acumulado para os conjuntos de 10, 30, e 50 LoRa-EDs por métodos de otimização e número de gateways.

com um número reduzido de episódios de aprendizado, o que denota a importância dessa solução. Considerando que foram mantidas a mesma granularidade, ou seja, mesmo número de posições a serem exploradas pelos VANTs durante o processo de aprendizado dos A2C-LoVQI e DQN-LoVQI para manter a comparação com o método de *baseline*. Ainda, dado que os algoritmos de aprendizado por reforço podem ser escalados para um número maior de posições, o que representa saltos menores em distância, pelos VANTs, a cada passo, assim, os resultados poderão alcançar outros posicionamentos ótimos que não foram investigados pelos *baselines*.

VI. Conclusões

Nesse artigo, aplicamos A2C para o problema de posicionamento de VANTs com objetivo de atender dispositivos IoT não-3GPP distribuídos em *slices*, mantendo os níveis de QoS dos acordos de acordos de níveis de serviço (*Service Level Agreement* – SLA). Propusemos dois algoritmos DRL para lidar com o problema, o DQN-LoVQI e o A2C-LoVQI. Nossos resultados demonstraram que é vantajoso adotar estratégias de DRL para solucionar essa classe de problemas e também mostrou que DQN-LoVQI e o A2C-LoVQI melhoraram consideravelmente os tempos de execução com garantia de cumprimento dos SLAs, e ainda demonstraram a viabilidade da proposta na solução dos problemas de escala e granularidade detectados nos algoritmos de otimização considerados como *baselines*. Disponibilizamos os nossos códigos publicamente nos repositórios GitHub[1,2]. A evolução natural desse trabalho está direcionada à inclusão de restrições inerentes à comunicação 3GPP das redes 5G e futuras, e.g., ondas milimétricas, além da integração 3GPP—não-3GPP. Por outra perspectiva, os trabalhos futuros também avançarão na adoção de estratégias aprimoradas de DRL e na ampliação da granularidade do espaço de busca visando explorar novos posicionamentos e possíveis melhorias de resultados.

Agradecimentos

Este trabalho foi apoiado em parte pela CAPES, MCTIC/CGI.br/FAPESP – Projeto Smart 5G Core And MULti-RAn Integration (SAMURAI) sob Concessão 2020/05127-2, CNPq – Projeto Universal sob Concessão 405111/2021-5, e RNP/MCTIC, Concessão nº 01245.010604/2020-14 – projeto Sistemas de Comunicações Móveis 6G.

Referências

[1] L. S. Vailshery, IoT connected devices worldwide 2019-2023, with forecasts to 2030, Statista, https://bit.ly/3Vyqdc9, 2023.
[2] B., Stefan, et al. *LPWAN in Context of 5G: Capab. of LoRaWAN to Contrib. mMTC.* IEEE 5th World Forum on IoT (WF-IoT). IEEE, 2019.
[3] LoRa Alliance Technical Committee. TS001-1.0.4 Lorawan® L2 1.0.4., https://bit.ly/3XdRQsg, Ac. (26/04/2020).
[4] Marchese, M., et al. *UAV and Satellite Employment for the Internet of Things Use Case*, Proc. IEEE Aerosp. Conf., pp. 1-8, 2020.
[5] Kirubakaran, B., et al. *Opt. UAV-Based Con. Solut. for Urban IoT Nets.* 15th Congr. on Ultra Modern Tel., Control Sys. Workshops. IEEE, 2023.
[6] Dawaliby, S. and Bradai, A. and Pousset, Y., Adaptive dynamic network slicing in LoRa networks, Future Generation Computer Systems, 2019.
[7] Dawaliby, S. and Bradai, A. and Pousset, Y.,Joint slice-based SF and TP optimization in LoRa smart city networks, Internet of Things, 2021.
[8] Tellache, A., et al., Proc. IEEE Int. Conf. Consum. Electron. (ICCE), DRL-based Resource Allocation in Dense Sliced LoRaWAN Net., 2022.
[9] Mardi, F. Z., et al., An Efficient Allocation System for Centralized NS in LoRaWAN, Proc. Wirel. Comm. Mob. Comput. Conf.(IWCMC), 2022.
[10] Marchese, Mario, et al., UAV and Satellite Employment for the Internet of Things Use Case,Proc. IEEE Aerosp. Conf., pp.1-8, 2020.
[11] Mahmood, M. et al., PSO-based joint UAV posit. & hybrid precoding in UAV-assisted massive MIMO systems, Proc. IEEE VTC, 2022.
[12] Almeida, E. N. et al., Traffic-Aware UAV Placement using a General DRL Method., Pr. IEEE Intern. Symp. Comp. Comm., 2022.
[13] Silva, R. S. et al. Dynamic resources allocation in non-3GPP IoT networks involving UAVs, Proc. IEEE Veh. Technol. Conf. (VTC), 2023.
[14] N. Parvaresh and B. Kantarci, A Contin. Actor-Critic DQL-Enabled Deploy. of UAV BSs, in IEEE Journal of the Comm. Society, 2023.
[15] G., Piotr and Z., Anatolij, ns-3 meets OpenAI Gym, ACM Conf. Model., Analysis and Simul. of Wireless and Mob. Systems (MSWiM), 2019
[16] Lee, D. , et al., IEEE Wireless Commun., Spatial modeling of the traffic density in cellular networks, pp.80–88, 2014
[17] Brasil, ANAC, Requisitos Gerais para Aeronaves Não Tripuladas para Uso Civil, Res. 419, 02/05/2017. [Emenda 02] de 1/06/2022.

[1]https://github.com/LABORA-INF-UFG/sliced-lorawan
[2]https://github.com/LABORA-INF-UFG/A2C-LoVQI