



UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO

ROGÉRIO SOUSA E SILVA

**Alocação dinâmica de recursos em
fatias de redes IoT não-3GPP
envolvendo VANTs**

Goiânia
2025



UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO (TECA) PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TESES

E DISSERTAÇÕES NA BIBLIOTECA DIGITAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio da Biblioteca Digital de Teses e Dissertações (BDTD/UFG), regulamentada pela Resolução CEPEC nº 832/2007, sem ressarcimento dos direitos autorais, de acordo com a [Lei 9.610/98](#), o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo das Teses e Dissertações disponibilizado na BDTD/UFG é de responsabilidade exclusiva do autor. Ao encaminhar o produto final, o autor(a) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do material bibliográfico

Dissertação Tese Outro*: _____

*No caso de mestrado/doutorado profissional, indique o formato do Trabalho de Conclusão de Curso, permitido no documento de área, correspondente ao programa de pós-graduação, orientado pela legislação vigente da CAPES.

Exemplos: Estudo de caso ou Revisão sistemática ou outros formatos.

2. Nome completo do autor

Rogério Sousa e Silva

3. Título do trabalho

Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs

4. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador)

Concorda com a liberação total do documento SIM NÃO¹

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante:

- a) consulta ao(à) autor(a) e ao(à) orientador(a);
 - b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo da tese ou dissertação.
- O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro;
- Publicação da dissertação/tese em livro.

Obs. Este termo deverá ser assinado no SEI pelo orientador e pelo autor.



Documento assinado eletronicamente por **Kleber Vieira Cardoso, Professor do Magistério Superior**, em 28/05/2025, às 15:29, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Rogério Sousa E Silva, Discente**, em 29/05/2025, às 12:31, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5401158** e o código CRC **9CBE654F**.

ROGÉRIO SOUSA E SILVA

Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs

Tese apresentada ao Programa de Pós Graduação em Ciência da Computação do Instituto de Informática da Universidade Federal de Goiás, como requisito para obtenção do título de Doutor em Ciência da Computação.

Área de concentração: Ciência da Computação.

Linha de pesquisa: Sistemas de Computação.

Orientador: Prof. Dr. Kleber Vieira Cardoso

Co-Orientador: Prof. Dr. Antonio Carlos de Oliveira Júnior

Goiânia
2025

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Sousa e Silva, Rogério

Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs [manuscrito] / Rogério Sousa e Silva. - 2025. CLXXXVI, 186 f.

Orientador: Prof. Kleber Vieira Cardoso; co-orientador Antonio Carlos de Oliveira Júnior.

Tese (Doutorado) - Universidade Federal de Goiás, Instituto de Informática (INF), , Goiânia, 2025.

Bibliografia. Apêndice.

Inclui abreviaturas, algoritmos, lista de figuras, lista de tabelas.

1. Veículos Aéreos Não Tripulados. 2. Programação Linear Inteira Mista. 3. Aprendizado por Reforço Profundo. 4. 5G não-3GPP. 5. Fatiamento de Redes não-3GPP. I. Vieira Cardoso, Kleber , orient. II. Título.

CDU 004



UNIVERSIDADE FEDERAL DE GOIÁS

INSTITUTO DE INFORMÁTICA

ATA DE DEFESA DE TESE

Ata nº **06/2025** da sessão de Defesa de Tese de **Rogério Sousa e Silva**, que confere o título de Doutor em Ciência da Computação, na área de concentração em Ciência da Computação.

Aos vinte e dois dias do mês de abril de dois mil e vinte e cinco, a partir das catorze horas, via webconferência, realizou-se a sessão pública de Defesa de Tese intitulada “**Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs**”. Os trabalhos foram instalados pelo Orientador, Professor Doutor Kleber Vieira Cardoso (INF/UFG) com a participação dos demais membros da Banca Examinadora: Professor Doutor Antonio Carlos de Oliveira Junior (INF/UFG), Coorientador; Professor Doutor Antônio Jorge Gomes Abelém (UFG), membro titular externo; Professor Doutor Cristiano Bonato Both (UNISINOS), membro titular externo; Professor Doutor Flávio Geraldo Coelho Rocha (EMC/UFG), membro titular externo; e Professor Doutor Leizer de Lima Pinto (INF/UFG), membro titular externo. A realização da banca ocorreu por meio de videoconferência. Durante a arguição os membros da banca não fizeram sugestão de alteração do título do trabalho. A Banca Examinadora reuniu-se em sessão secreta a fim de concluir o julgamento da Tese, tendo sido o candidato **aprovado** pelos seus membros. Proclamados os resultados pelo Professor Doutor Kleber Vieira Cardoso, Presidente da Banca Examinadora, foram encerrados os trabalhos e, para constar, lavrou-se a presente ata que é assinada pelos Membros da Banca Examinadora, aos vinte e dois dias do mês de abril de dois mil e vinte e cinco.

TÍTULO SUGERIDO PELA BANCA



Documento assinado eletronicamente por **CRISTIANO BONATO BOTH, Usuário Externo**, em 22/04/2025, às 18:01, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Antonio Carlos De Oliveira Junior, Professor do Magistério Superior**, em 22/04/2025, às 18:02, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Kleber Vieira Cardoso, Professor do Magistério Superior**, em 22/04/2025, às 18:06, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Rogério Sousa E Silva, Discente**, em 22/04/2025, às 18:07, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Antonio Jorge Gomes Abelem, Usuário Externo**, em 22/04/2025, às 18:08, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Leizer De Lima Pinto, Professor do Magistério Superior**, em 22/04/2025, às 18:09, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Flavio Geraldo Coelho Rocha, Professor do Magistério Superior**, em 22/04/2025, às 18:54, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5310435** e o código CRC **468B90CB**.

Referência: Processo nº 23070.011018/2025-75

SEI nº 5310435

ROGÉRIO SOUSA E SILVA

Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs

Tese defendida no Programa de Pós-Graduação do Instituto de Informática da Universidade Federal de Goiás como requisito parcial para obtenção do título de Doutor em Ciência da Computação, aprovada em 01 de Fevereiro de 2025, pela Banca Examinadora constituída pelos professores:

Prof. Dr. Kleber Vieira Cardoso

Instituto de Informática – UFG

Presidente da Banca

Prof. Dr. Antonio Carlos de Oliveira Júnior

Instituto de Informática – UFG

Prof. Dr. Antônio Jorge Gomes Abelém

Universidade Federal do Pará – UFPA

Prof. Dr. Cristiano Bonato Both

Pós-Graduação em Computação Aplicada - UNISINOS

Prof. Dr. Leizer de Lima Pinto

Instituto de Informática – UFG

Prof. Dr. Flávio Geraldo Coelho Rocha

Escola de Engenharia Elétrica, Mecânica e de Computação – UFG

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador(a).

Rogério Sousa e Silva

É Técnico em Processamento de Dados pelo Instituto Unificado de Ensino Superior Objetivo (1993). Recebeu o título de Mestre em Ciência da Computação pela Universidade Federal de Uberlândia (2007), desenvolvendo a dissertação "Distribuição de Tarefas em Sistemas de Workflow por meio da Seleção Induzida de Recursos". Tem experiência como docente e pesquisador nas áreas de computação vinculados ao Instituto Federal de Goiás (IFG). Possui interesse em projetos de pesquisa relacionados à área de Ciências da Computação, com ênfase em redes móveis futuras, internet das coisas e cidades inteligentes, computação paralela e distribuída, sistemas inteligentes, e sistemas aplicados à eficiência energética e sustentabilidade.

À Greyce e à Isabela, que me amam incondicionalmente.

À minha mãe D. Dilma, e aos meus irmãos Roberto e Fábio, minha família e aos meus amigos, que me dedicam todo o apoio e carinho necessários para uma vida feliz e completa.

Agradecimentos

Greyce Revoredo e Isabela Revoredo Sousa, obrigado por lutarem ao meu lado e sempre acreditarem. Obrigado pelo amor incondicional. As AMO demais!

Kleber e Antonio, obrigado por serem muito mais que orientadores. O apoio e dedicação de vocês me guiou pelas veredas do conhecimento. Meus grandes Amigos!

Ao Instituto Federal de Goiás, ao Campus Inhumas, e a todos os meus colegas de luta. Obrigado pelo apoio fundamental para essa conquista.

Ao NumbERS e a todos os nossos alunos. Obrigado pela motivação para evoluir sempre.

Ao professor Leandro Alexandre Freitas, meu grande Amigo, obrigado por estar comigo nessa caminhada, por acreditar e me apoiar sempre. #TMJ

Aos meus babies, Estrela, Princesa Serafine, Jhonny, Menina†, Yordan†, Delícia, Fuzão† e Fuzinho, Mia, Sushi, Pãozinho e Mexerica. Vocês são a alegria de todos os dias.

<If in a dark state we find ourselves... a little more knowledge might light
our way.>

<Yoda >,
<*STAR WARS: Ep. III – Revenge of the Sith, 2005* >.

Resumo

Silva, Rogério S.. **Alocação dinâmica de recursos em fatias de redes IoT não-3GPP envolvendo VANTs**. Goiânia, 2025. 188p. Tese de Doutorado. Instituto de Informática, Universidade Federal de Goiás.

O crescimento exponencial da Internet das Coisas impõe desafios crescentes à infraestrutura de comunicação, especialmente em cenários críticos, como desastres naturais e eventos com alta densidade populacional, nos quais a sobrecarga de rede compromete a continuidade e a qualidade dos serviços. Nesse contexto, esta tese apresenta uma abordagem híbrida para a alocação dinâmica de recursos em redes IoT não-3GPP, integrando tecnologias de fatiamento de rede, acesso heterogêneo e veículos aéreos não tripulados. A abordagem híbrida desenvolvida combina, de forma sinérgica, a precisão dos métodos de otimização exata baseados em Programação Linear Inteira Mista, empregados para o posicionamento inicial ótimo dos VANTs, com a flexibilidade adaptativa de algoritmos avançados de Aprendizado por Reforço Profundo, responsáveis pelo reposicionamento dinâmico e autônomo em ambientes variáveis. A primeira consiste na formulação de modelos exatos para determinar o posicionamento inicial ótimo dos VANTs, com o objetivo de minimizar custos e maximizar os níveis de Qualidade de Serviço. Na segunda, é aplicada uma abordagem adaptativa, permitindo decisões autônomas de reposicionamento frente à dinâmica do ambiente e às variações nas demandas da rede. Quatro algoritmos foram desenvolvidos e avaliados, e.g., SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO. As soluções propostas foram validadas por meio do simulador de redes ns-3, em cenários personalizados com conectividade não-3GPP. Os resultados demonstraram ganhos expressivos na QoS, redução do número de VANTs implantados, maior robustez das decisões e aumento da eficiência espectral, com destaque para os algoritmos NSE-A2C e RG2E-PPO. A abordagem híbrida apresentada promove uma infraestrutura de comunicação móvel, escalável e resiliente, capaz de responder de forma autônoma e eficiente às demandas específicas de diferentes aplicações IoT. Esta tese contribui para o estado da arte ao propor uma arquitetura híbrida replicável, sustentável e orientada a serviços, para comunicação confiável em redes heterogêneas e dinâmicas baseadas em tecnologias não licenciadas. Suas aplicações incluem cidades inteligentes, resposta a desastres e implantação temporária de conectividade em cenários de infraestrutura degradada ou inexistente.

Palavras-chave

Veículos Aéreos Não Tripulados, Programação Linear Inteira Mista, Aprendizado por Reforço Profundo, Redes 5G não-3GPP, Fatiamento de Redes.

Abstract

Silva, Rogério S.. **Dynamic Resource Allocation in Non-3GPP IoT Network Slices Involving UAVs**. Goiânia, 2025. 188p. PhD. Thesis. Instituto de Informática, Universidade Federal de Goiás.

The exponential growth of the Internet of Things (IoT) has introduced increasing challenges to communication infrastructures, particularly in critical scenarios such as natural disasters and densely populated events, where network overload compromises service continuity and quality. In this context, this thesis presents a hybrid approach for dynamic resource allocation in non-3GPP IoT networks, integrating network slicing (NS), heterogeneous access (Multi-RAT), and unmanned aerial vehicles (UAVs) equipped with LoRaWAN gateways. The proposed hybrid approach synergistically combines the precision of exact optimization methods based on Mixed Integer Linear Programming (MILP), employed for determining the optimal initial positioning of UAVs, with the adaptive flexibility of advanced Deep Reinforcement Learning (DRL) algorithms, which enable dynamic and autonomous repositioning in variable environments. The goal of the first stage is to minimize operational and deployment costs while maximizing Quality of Service (QoS), while the second stage is to facilitate the autonomous repositioning of UAVs in response to environmental changes and fluctuations in network demand. We develop and assess four DRL algorithms, e.g., SR-DQN, DA-DDDQN, NSE-A2C, and RG2E-PPO. The proposed solutions were validated through realistic simulations using the ns-3 network simulator, in customized scenarios with non-3GPP connectivity. Results demonstrated significant improvements in QoS, reduced number of deployed UAVs, enhanced decision robustness, and increased spectral efficiency, with notable performance from the NSE-A2C and RG2E-PPO algorithms. The hybrid approach enables the creation of a mobile, scalable, and resilient communication infrastructure capable of autonomously and efficiently addressing the specific requirements of diverse IoT applications, particularly in urban and emergency environments with critical connectivity constraints. This thesis contributes to the state of the art by proposing a replicable, sustainable, and service-oriented hybrid architecture for reliable communication in heterogeneous and dynamic networks based on unlicensed technologies. Potential applications include smart cities, disaster response, and temporary connectivity deployment in degraded or non-existent infrastructure scenarios.

Keywords

Unmanned Aerial Vehicles, Mixed Integer Linear Programming, Deep Reinforcement Learning, Non-3GPP 5G Networks, Non-3GPP Network Slicing.

Sumário

Lista de Acrônimos	16
Lista de Algoritmos	19
Lista de Figuras	20
Lista de Tabelas	22
1 Introdução	23
1.1 Escopo da Tese	27
1.2 Questões de Pesquisa	29
1.3 Organização do trabalho	31
2 Referencial teórico	33
2.1 Redes não-3GPP, Fatiamento e Multi-RAT	33
2.2 LoRa e LoRaWAN	34
2.3 <i>Network Slicing</i>	40
2.4 NS em LoRaWAN	43
2.5 Veículos Aéreos não Tripulados	45
2.6 Aprendizado de Máquina	46
2.7 Conclusão	56
3 Trabalhos Relacionados	58
3.1 Redes Não-3GPP e <i>Network Slicing</i>	60
3.2 VANTs como parte da infraestrutura de comunicação	62
3.3 Análise Crítica, Lacunas e Comparações	68
3.4 Conclusão	69
4 Abordagem Exata	71
4.1 Evolução da pesquisa com MILP	71
4.2 Modelagem do Sistema	72
4.3 Formulação do Problema	74
4.4 Avaliações	79
4.5 Resultados	87
4.6 Conclusão	104

5	Abordagem Baseada em Aprendizado de Máquina	105
5.1	Modelagem <i>Markov Decision Process</i> (MDP)	106
5.2	Formulação do Problema	111
5.3	Avaliações	121
5.4	Resultados	134
5.5	Conclusão	152
6	Considerações Finais e Trabalhos Futuros	154
6.1	Resultados Obtidos	155
6.2	Trabalhos Futuros	158
	Referências Bibliográficas	161
A	Uso de Ferramentas de Inteligência Artificial	172
A.1	Canva Dream-Lab.	172
A.2	OpenAI ChatGPT (GPT-4o)	175
A.3	Assistente de código JetBrains AI Assistant	178

Lista de Acrônimos

Português:

ANAC Agência Nacional de Aviação Civil
IA Inteligência Artificial
RBAC-E Regulamento Brasileiro da Aviação Civil Especial
VANT Veículo Aéreo Não Tripulado

Inglês:

3GPP *3rd Generation Partnership Project*
5G *5th Generation Mobile Networks*
5GC *5G Core*
5GS *5G System*
6G *6th Generation Mobile Networks*
A2C *Advantage Actor Critic*
AC *Actor Critic*
ACDQL *Actor-Critic Deep Q-learning*
ADR *Adaptive Data Rate*
API *Application Programming Interface*
BS *Base Station*
BW *Bandwidth*
CAPEX *Capital Expenditure*
CDMADDPG *Central and Distributed Multi-Agent Deep Deterministic Policy Gradient*
CF *Carrier Frequency*
COW *Cell-On-Wheels*
CP *Control Plane*
CR *Code Rate*
CRC *Cyclic Redundancy Check*
CSS *Chirp Spread Spectrum*
D2D *Device to Device*
DDQN *Double Deep Q-Network*
DA-DDDQN *Domain-Aware Double Dueling Deep Q-Network*
DDPG *Deep Deterministic Policy Gradient*
DFQL *Deep Federated Q-Learning*
DNN *Deep Neural Network*
DP *Data Plane*
DR *Data Rate*
DRL *Deep Reinforcement Learning*
DQL *Deep Q-Learning*
DQN *Deep Q-Network*

DSC *Drone Small Cell*
ED *End Device*
FedAVG *Federated Averaging*
FL *Federated Learning*
FSK *Frequency Shifting Keying*
GA *Genetic Algorithms*
GAE *Generalized Advantage Estimator*
GMM *Geometric Mean Method*
IoT *Internet of Things*
IIoT *Industrial Internet of Things*
ISM *Industrial, Scientific and Medical*
ITU-T *International Telecommunication Union - Telecomm. Standardization Sector*
LDPL *Log-Distance Path Loss*
LoRa *Long Range*
LoRa-ED *LoRa End Devices*
LoRaWAN *Long Range Wide Area Network*
LoS *Line Of Sight*
LPWAN *Low Power Wide Area Network*
LTE-M *Long Term Evolution for Machines*
MAB *Multi-armed Bandit*
MAC *Media Access Control Layer*
MDP *Markov Decision Process*
MEC *Multi-Access Edge Computing*
MHQ-PRP *Multi-hop Queuing Model with M/G/1 Preemptive Repeat Priority*
MBGD *Mini-Batch Gradient Descent*
MILP *Mixed Integer Linear Programming*
ML *Machine Learning*
MLE *Maximum Likelihood Estimation*
mmWave *Millimeter Wave*
MOMDP *Multi-Objective Markov Decision Process*
MORL-TER *Multi-Objective Reinforcement Learning Trace-based Experience Replay*
MQTT *Message Queuing Telemetry Transport*
MSE *Mean Square Error*
Multi-RAT *Multi-Radio Access Technology*
NB-IoT *Narrowband-IoT*
NF *Network Function*
NFV *Network Function Virtualization*
NGMN *Next-Generation Mobile Networks*
NN *Neural Network*
NOMA *Non-Orthogonal Multiple Access*
NS *Network Slicing*
ns-3 *Network Simulator 3*
NSE-A2C *Normalized-State Entropy Advantage Actor-Critic*
NSI *Network Slice Instance*
NST *Network Slice Template*
NTN *Non-Terrestrial Network*
NT-BS *Non-Terrestrial Base Station*

NT-UE *Non-Terrestrial User Equipment*
OGMMC *Online Gaussian Mixture Model for clustering*
OPEX *Operating Expenditure*
PDR *Packet Drop Rate*
PER *Prioritized Experience Replay*
PFL *Personalized Federated Learning*
PFDRL *Personalized Federated Deep Reinforcement Learning*
PHY *Physical Layer*
PPO *Proximal Policy Optimization*
PSO *Particle Swarm Optimization*
QL *Q-Learning*
QoS *Quality of Service*
RAN *Radio Access Network*
ReLU *Rectified Linear Unit*
RG2E-PPO *Reward-Guided and Generalized-Advantage Entropic Proximal Policy Optimization*
RL *Reinforcement Learning*
RSSI *Received Signal Strength Indicator*
SAn *Simulated Annealing*
SAGIN *Space-Air-Ground Integrated Network*
SBA *Service-Based Architecture*
SCA *Successive Convex Approximation*
SDN *Software Defined Network*
SF *Spreading Factor*
SINR *Signal-to-Interference-plus-Noise Ratio*
SLA *Service Level Agreement*
SNR *Signal-to-Noise Ratio*
SR-DQLearning *k-step State Reduction Distributive Q-learning*
SR-QLearning *k-step State Reduction Q-learning*
SR-DQN *Shaped-Reward Deep Q-Network*
SS-RSRP *Synchronization Signal - Reference Signal Received Power*
SS-SINR *Synchronization Signal - Signal-to-Interference-plus-Noise Ratio*
TD *Temporal Difference*
ToA *Time-on-Air*
TOPSIS *Technique for Order of Preference by Similarity to Ideal Solution*
TP *Transmission Power*
TRPO *Trust Region Policy Optimization*
TUPA *Traffic-aware UAVs Placement Algorithm*
UAV-BS *Unmanned Aerial Vehicle-mounted Base Station*
UMa *Urban Macro*
VNF *Virtualized Network Function*
ZMQ *ZeroMQ Messaging Library*

Lista de Algoritmos

- 5.1 *Shaped-Reward Deep Q-Network (SR-DQN)* para Alocação Dinâmica de Recursos em Redes IoT com VANTs 113
- 5.2 *Domain-Aware Double Dueling Deep Q-Network (DA-DDDQN)* para Alocação Dinâmica de Recursos em Redes IoT com VANT 116
- 5.3 *Normalized-State Entropy Advantage Actor-Critic (NSE-A2C)* para Alocação Dinâmica de Recursos em Redes IoT com VANT. [Criada pelo Autor.] 118
- 5.4 *Reward-Guided and Generalized-Advantage Entropic Proximal Policy Optimization (RG2E-PPO)* para Alocação Dinâmica de Recursos em Redes IoT com VANT 120

Lista de Figuras

1.1	Cenário para o fatiamento de redes IoT não-3GPP envolvendo VANTs para suporte a surtos de demanda. [Desenvolvida pelo Autor.]	27
1.2	Exemplos de aplicação para suporte a surtos de demanda em diferentes cenários: eventos musicais, esportivos, deslizamentos de encostas e incêndios florestais. [Imagens geradas por IA com o uso do <i>Canva Dream Lab</i> , conforme descrito no Apêndice A.1]	28
2.1	Protocolo LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [LoRa Alliance 2015].]	35
2.2	ToA, taxa de dados e consumo energético em relação ao SF. [Desenvolvida pelo Autor. Adaptada de [Semtech 2019]]	38
2.3	Componentes da arquitetura LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [Semtech 2019]]	39
2.4	Ciclo de vida de um <i>Slice</i> . [Desenvolvida pelo Autor. Adaptada de [3GPP 2020]]	43
2.5	Arquitetura de <i>network slicing</i> em LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [Dawaliby, Bradai e Pousset 2019]]	44
2.6	Interação entre o Agente e o Ambiente em um MDP. [Desenvolvida pelo Autor. Adaptada de [Sutton e Barto 2018]]	47
2.7	Diagrama de estados e transições para um MDP. Número de estados e transições arbitrários. [Diagrama produzido por IA conforme descrito no Apêndice A.2]	48
2.8	Estrutura típica de uma NN com camada de entrada, camadas ocultas e camada de saída. Número de nós e camadas arbitrários para melhor representação. [Criado pelo Autor. Diagrama produzido em: http://alexlenail.me/NN-SVG/index.html .]	50
4.1	Metodologia do processo de avaliação dos métodos de otimização propostos. [Criada pelo Autor.]	79
4.2	Ilustração da distribuição log-normal de dispositivos em uma área de 20 km ² . [Criada pelo Autor.]	81
4.3	Ilustração das posições candidatas para implantação de VANTs. [Criada pelo Autor.]	82
4.4	Otimizações realizadas por número de dispositivos e posições candidatas. [Criada pelo Autor.]	83
4.5	Comparação do tempo de execução das otimizações para os solucionadores Gurobi e SCIP. [Criada pelo Autor.]	85
4.6	Tempo de execução em função do número de posições candidatas para as estratégias experimentadas. [Criada pelo Autor.]	88

4.7	Média de QoS em função do número de posições candidatas, considerando otimizações e simulações no ns-3. O limiar mínimo foi fixado em 0,9. [Criada pelo Autor.]	90
4.8	Média de QoS em função do número de dispositivos para as estratégias experimentadas. [Criada pelo Autor.]	92
4.9	Distribuição percentual de dispositivos por SF para cada estratégia de otimização. [Criada pelo Autor.]	94
4.10	Desempenho das estratégias em função do número de dispositivos. [Criada pelo Autor.]	95
4.11	Desempenho das estratégias em função do número de posições candidatas. [Criada pelo Autor.]	95
4.12	Distribuição da QoS por faixas de VANTs implantados, considerando diferentes estratégias de otimização e simulação. [Criada pelo Autor.]	97
4.13	Número médio de VANTs implantados em função do número de dispositivos para as diferentes estratégias na otimização. [Criada pelo Autor.]	99
4.14	Interferência média em função do número de posições candidatas. [Criada pelo Autor.]	101
4.15	Interferência média em função do número de dispositivos. [Criada pelo Autor.]	102
5.1	Crescimento do espaço de estados $ S $ em função do número de posições candidatas $ P $ e do número de VANTs δ . [Criada pelo Autor.]	107
5.2	Gráfico do crescimento de $ A $ em função de δ . [Criada pelo Autor.]	108
5.3	Metodologia do processo de treinamento e avaliação das soluções baseadas em aprendizado de máquina. [Criada pelo Autor.]	123
5.4	Desempenho do SR-DQN para 2, 3, 4 e 5 VANTs. [Criada pelo Autor.]	135
5.5	Desempenho do DA-DDDQN para 2, 3, 4 e 5 VANTs. [Criada pelo Autor.]	137
5.6	Impacto da Variação no Número de VANTs no Desempenho do NSE-A2C. [Criada pelo Autor.]	140
5.7	Impacto da Variação na Quantidade de Dispositivos no Desempenho do NSE-A2C. [Criada pelo Autor.]	140
5.8	Impacto da variação no número de VANTs no desempenho do RG2E-PPO. [Criada pelo Autor.]	144
5.9	Impacto da variação na quantidade de dispositivos no desempenho do RG2E-PPO. [Criada pelo Autor.]	144
5.10	Variações nos tempos médios de execução dos algoritmos para 2 e 4 VANTs e 100 dispositivos. [Criada pelo Autor.]	147
5.11	PDR para diferentes algoritmos de DRL em função do número de posições candidatas (à esquerda) e do número de dispositivos (à direita). [Criada pelo Autor.]	149
5.12	Evolução da QoS ao longo dos episódios para os algoritmos NSE-A2C e RG2E-PPO. [Criada pelo Autor.]	152
A.1	Resultados para o comando: “Evento musical”.	173
A.2	Resultados para o comando: “Evento Esportivo”.	174
A.3	Resultados para o comando “Deslizamento de encosta”.	174
A.4	Resultados para o comando “Incêndio na zona rural”.	175

Lista de Tabelas

2.1	DR segundo valores usuais de SF, BW e CR. [Desenvolvida pelo Autor.]	37
2.2	Duração de um símbolo em relação ao SF e BW. [Criada pelo Autor.]	37
2.3	Síntese das normas de utilização de VANTs no Brasil e na Europa, [Desenvolvida pelo Autor.]	46
3.1	Características dos Trabalhos Relacionados. [Desenvolvida pelo Autor.]	59
4.1	Notação utilizada na formalização da solução por MILP. [Criada pelo Autor.]	78
4.2	Parâmetros utilizados nas simulações no ns-3. [Criada pelo Autor.]	84
4.3	Parâmetros utilizados na otimização. [Criada pelo Autor.]	86
5.1	Lista de símbolos adotados na descrição do MDP, organizados por categoria. [Criada pelo Autor.]	110
5.2	Hiperparâmetros utilizados no treinamento do SR-DQN. [Criada pelo Autor.]	125
5.3	Hiperparâmetros utilizados no treinamento do DA-DDDQN. [Criada pelo Autor.]	126
5.4	Hiperparâmetros utilizados no treinamento do NSE-A2C. [Criada pelo Autor.]	129
5.5	Hiperparâmetros utilizados no treinamento do RG2E-PPO. [Criada pelo Autor.]	130
5.6	Indicadores médios por quantidade de VANTs após 300 episódios para o algoritmo SR-DQN. [Criada pelo Autor.]	136
5.7	Indicadores médios por quantidade de VANTs após 300 episódios para o algoritmo DA-DDDQN. [Criada pelo Autor.]	139
5.8	Resumo quantitativo do desempenho do NSE-A2C com variação no número de VANTs [Valores estimados a partir das curvas médias]. [Criada pelo Autor.]	142
5.9	Resumo dos resultados para diferentes quantidades de dispositivos com NSE-A2C. [Criada pelo Autor.]	142
5.10	Resumo quantitativo do desempenho do RG2E-PPO com variação no número de VANTs. Valores estimados a partir das curvas médias. [Criada pelo Autor.]	146
5.11	Resumo dos resultados para diferentes quantidades de dispositivos com RG2E-PPO. [Criada pelo Autor.]	146
5.12	Resumo das limitações computacionais dos algoritmos baseados em função de valor e seus impactos na avaliação. [Desenvolvido pelo Autor.]	150
6.1	Publicações no tema da tese.[Criada pelo Autor.]	156
6.2	Publicações alinhadas ao tema da tese. [Criada pelo Autor.]	157
6.3	Software e Repositórios de Código. [Criada pelo Autor.]	158

Introdução

Eventos ocasionais de grande porte, e.g., manifestações políticas, festivais musicais ou competições esportivas, geram aglomerações que frequentemente sobrecarregam a infraestrutura de comunicação, resultando em degradação acentuada da Qualidade de Serviço (*Quality of Service (QoS)*) e, em situações críticas, na interrupção completa dos serviços. Paralelamente, a intensificação de desastres naturais, e.g., alagamentos e deslizamentos, agravada pelas mudanças climáticas, tem comprometido de forma significativa a estabilidade das redes de comunicação. A ausência de conectividade nesses cenários impacta não apenas a continuidade dos serviços, mas também coloca em risco a integridade de populações vulneráveis, dificultando operações de resgate e a prestação de assistência emergencial [[Horizonte 2022](#), [Guardian 2023](#), [Boosters 2023](#), [InfoMoney 2024](#), [PPIC 2023](#)].

Diante desses desafios, é essencial desenvolver soluções tecnológicas avançadas que possam atender às demandas emergenciais de comunicação, acompanhar o crescimento exponencial da *Internet of Things (IoT)* e lidar com a crescente necessidade de conectividade. Essas soluções devem ser capazes não apenas de restabelecer a comunicação em situações adversas, mas também de oferecer escalabilidade e eficiência para suportar eventos de alta demanda e a proliferação de dispositivos conectados. Inovações como o *Network Slicing (NS)* e as tecnologias de múltiplas redes de acesso (*Multi-Radio Access Technologies (Multi-RATs)*) estão se consolidando como alternativas promissoras, proporcionando flexibilidade, eficiência e suporte a uma ampla gama de aplicações, especialmente aquelas voltadas à *IoT*.

Observa-se a consolidação da *IoT* como uma das tecnologias mais disruptivas da atualidade, conectando bilhões de dispositivos em diversos contextos, desde aplicações industriais até cidades inteligentes [[Marchese, Moheddine e Patrone 2019](#)]. Esse crescimento exponencial tem revolucionado a maneira como os usuários interagem com redes de comunicação, mas também apresenta desafios significativos. Cenários emergenciais ou de alta densidade populacional, frequentemente expõem as limitações das infraestruturas de rede tradicionais, que não conseguem atender às demandas crescentes. Embora a maioria dos dispositivos conectados ainda dependa de redes sem fio para comunicação,

a infraestrutura atual enfrenta dificuldades para suportar o rápido aumento das conexões de **IoT**, somadas às conexões móveis convencionais. Além disso, essas redes geralmente falham em satisfazer os diversos requisitos de **QoS** impostos pelos diferentes cenários de aplicação da **IoT** [Wijethilaka e Liyanage 2021].

Trabalhos recentes [Wijethilaka e Liyanage 2021, Sundaram, Du e Zhao 2020] exploraram soluções para mitigar as limitações das redes convencionais em atender demandas emergenciais e de alta densidade populacional. No entanto, essas propostas são, em geral, limitadas a contextos estáticos ou específicos de uma única tecnologia, carecendo de abordagens que integrem mobilidade, flexibilidade e interoperabilidade em ambientes heterogêneos. Para superar essas restrições, recursos como o **NS**, a integração com redes não-*3rd Generation Partnership Project* (**3GPP**) e o uso de **Multi-RATs** permitem o desenvolvimento de soluções mais robustas, escaláveis e adaptáveis, oferecendo conectividade confiável e eficiente para atender aos requisitos complexos de **IoT** em cenários críticos e de alta densidade. Essa evolução tecnológica pavimenta o caminho para propostas inovadoras, como a apresentada neste trabalho, que explora a combinação de Veículos Aéreos Não Tripulados (**VANTs**) equipados com *gateways Long Range Wide Area Network* (**LoRaWAN**) com o suporte de redes não-3GPP **LoRaWAN**, viabilizando infraestruturas resilientes e dinâmicas.

As redes não-**3GPP** têm ganhado destaque na composição de soluções de conectividade para cenários emergenciais e de alta densidade, especialmente por sua capacidade de operar de forma independente ou integrada a arquiteturas 3GPP. Dentre essas tecnologias, as redes de longa distância e baixa potência (*Low Power Wide Area Networks* (**LPWANs**)), como *Narrowband-IoT* (**NB-IoT**), *Long Term Evolution for Machines* (**LTE-M**) e, especialmente, o **LoRaWAN**, desempenham um papel central no suporte à conectividade de dispositivos **IoT** em aplicações que demandam baixo consumo energético e cobertura geográfica ampliada [Sundaram, Du e Zhao 2020]. Essas redes são particularmente eficazes em aplicações urbanas, como monitoramento ambiental, redes de sensores distribuídos e automação de infraestrutura, permitindo escalabilidade com baixo custo.

Para garantir a interoperabilidade entre diferentes tecnologias de acesso, as arquiteturas modernas têm adotado o paradigma **Multi-RAT**, viabilizando a integração entre redes **3GPP** e não-**3GPP** [3GPP 2022]. Essa integração permite que dispositivos operem de forma transparente em ambientes heterogêneos, combinando o alcance e a eficiência das redes **LPWANs** com a capacidade de transmissão das redes celulares tradicionais. Tal característica é essencial em cenários críticos, nos quais a flexibilidade da infraestrutura pode determinar a continuidade dos serviços de comunicação.

Nesse contexto, o **NS** emerge como uma tecnologia-chave para viabilizar a personalização e o isolamento lógico de recursos em redes compartilhadas. Em espe-

cial, o fatiamento de redes não-3GPP, como o LoRaWAN, permite a criação de fatias dedicadas com parâmetros operacionais distintos, e.g., *Spreading Factor (SF)*, potência de transmissão e requisitos de latência, adequados a diferentes perfis de aplicações [Dawaliby et al. 2018]. Essa abordagem assegura que aplicações críticas, como telemetria em tempo real ou controle de infraestrutura, recebam a QoS necessária, ao mesmo tempo em que compartilham a mesma infraestrutura física com aplicações de menor criticidade, de maneira otimizada [Popovski et al. 2018].

Apesar dessas inovações, cenários emergenciais ou eventos de alta densidade populacional podem demandar uma infraestrutura ainda mais adaptável e dinâmica, capaz de expandir rapidamente a cobertura e aliviar gargalos em situações críticas. Nesse contexto, os VANTs surgem como uma solução complementar promissora, desempenhando um papel crucial na ampliação temporária e direcionada da infraestrutura de comunicação. Embora as redes móveis de quinta geração (*5th Generation Mobile Networks (5G)*) desempenhem um papel complementar ao oferecer suporte arquitetural para integração e virtualização por meio de interfaces padronizadas e funções softwarizadas, o foco deste trabalho recai sobre a orquestração de redes não-3GPP com suporte ao NS, empregando VANTs como elementos móveis para alocação dinâmica de recursos. Essa estratégia proporciona maior resiliência, adaptabilidade e redução de custos operacionais, essenciais para a construção de soluções escaláveis em cenários de demanda elevada e infraestrutura degradada.

VANTs equipados com *gateways* LoRaWAN, podem ser implantados sob demanda para atender surtos de comunicação, conectando dispositivos IoT por meio de enlaces *Long Range (LoRa)* [Pires-Jr et al. 2024, Silva et al. 2024]. A escolha de VANTs como elementos de infraestrutura complementar deve-se à sua alta mobilidade, flexibilidade de implantação e custo reduzido em comparação com infraestruturas fixas. Além disso, tecnologias como LoRaWAN se destacam em aplicações IoT devido ao seu baixo consumo energético e capacidade de comunicação de longo alcance. Em relação a alternativas como NB-IoT ou LTE-M, o LoRaWAN oferece vantagens para cenários emergenciais, como suporte eficiente em faixas de frequência não licenciadas e fácil integração com dispositivos existentes. Essas características tornam a combinação de VANTs e LoRaWAN particularmente atraente para situações de sobrecarga de infraestrutura ou falhas críticas.

Essa abordagem oferece uma solução móvel e escalável para lidar com cenários desafiadores, possibilitando elevar os níveis de QoS mesmo em condições adversas. Dessa forma, os VANTs se consolidam como uma peça-chave na construção de uma infraestrutura de comunicação resiliente e adaptável em cenários críticos. Além de garantir conectividade em situações de crise, a utilização de VANTs equipados com *gateways* LoRaWAN pode trazer benefícios econômicos e sociais significativos. A redução de cus-

tos operacionais, pela substituição de infraestrutura fixa por dispositivos móveis, pode viabilizar a implementação em regiões economicamente desfavorecidas. Em termos sociais, a conectividade emergencial possibilita uma resposta mais rápida em operações de resgate e salvamento, reduzindo o impacto de desastres e potencialmente salvando vidas em situações críticas.

Diante dos desafios impostos por cenários de alta densidade e emergência, esta tese propõe uma abordagem híbrida inovadora para a alocação dinâmica de recursos em redes IoT não-3GPP, por meio do uso de VANTs equipados com gateways LoRaWAN e com suporte ao fatiamento de redes (NS). A proposta metodológica é estruturada em duas etapas complementares, que integram técnicas de otimização exata e aprendizado por reforço profundo para promover soluções adaptativas, escaláveis e eficientes.

Na primeira etapa, é formulado um modelo de otimização baseado em *Mixed Integer Linear Programming* (MILP), com o objetivo de determinar a posição inicial ótima dos VANTs, minimizando os custos de implantação e maximizando os níveis de QoS. Essa formulação opera sobre um espaço de busca discretizado com baixa granularidade, viabilizando soluções determinísticas e computacionalmente viáveis.

A segunda etapa expande essa solução inicial ao incorporar técnicas de *Deep Reinforcement Learning* (DRL), que permitem o reposicionamento dinâmico e autônomo dos VANTs em cenários com granularidade aumentada. O problema é modelado como um MDP, possibilitando o desenvolvimento de quatro algoritmos originais, i.e., *Shaped-Reward Deep Q-Network* (SR-DQN), *Domain-Aware Double Dueling Deep Q-Network* (DA-DDDQN), *Normalized-State Entropy Advantage Actor-Critic* (NSE-A2C) e *Reward-Guided and Generalized-Advantage Entropic Proximal Policy Optimization* (RG2E-PPO), capazes de adaptar a configuração da rede às variações espaciais e temporais do ambiente. Os experimentos, conduzidos por meio de simulações no ambiente *Network Simulator 3* (ns-3), evidenciam ganhos expressivos em termos de escalabilidade, cobertura, confiabilidade e redução do número de VANTs necessários, impactando diretamente na diminuição de *Capital Expenditure* (CAPEX) e *Operating Expenditure* (OPEX).

Como principal contribuição, esta tese apresenta uma solução híbrida que combina o rigor da otimização exata com a adaptabilidade do aprendizado por reforço, oferecendo uma alternativa robusta e eficaz para a orquestração de infraestruturas móveis e resilientes em redes IoT não-3GPP. As contribuições geradas foram amplamente disseminadas por meio de publicações científicas, capítulo de livro, minicurso e repositórios de código-fonte abertos, ampliando seu impacto acadêmico e prático.

1.1 Escopo da Tese

Este trabalho aborda os desafios relacionados à sobrecarga e interrupções na infraestrutura de comunicação em cenários críticos, tais como eventos de alta densidade populacional e desastres naturais. A proposta combina **VANTs** equipados com *gateways* **LoRaWAN**, redes **IoT** não-**3GPP** e **NS** para formar uma arquitetura adaptável e resiliente, projetada para responder de maneira eficiente a surtos de demanda por conectividade.

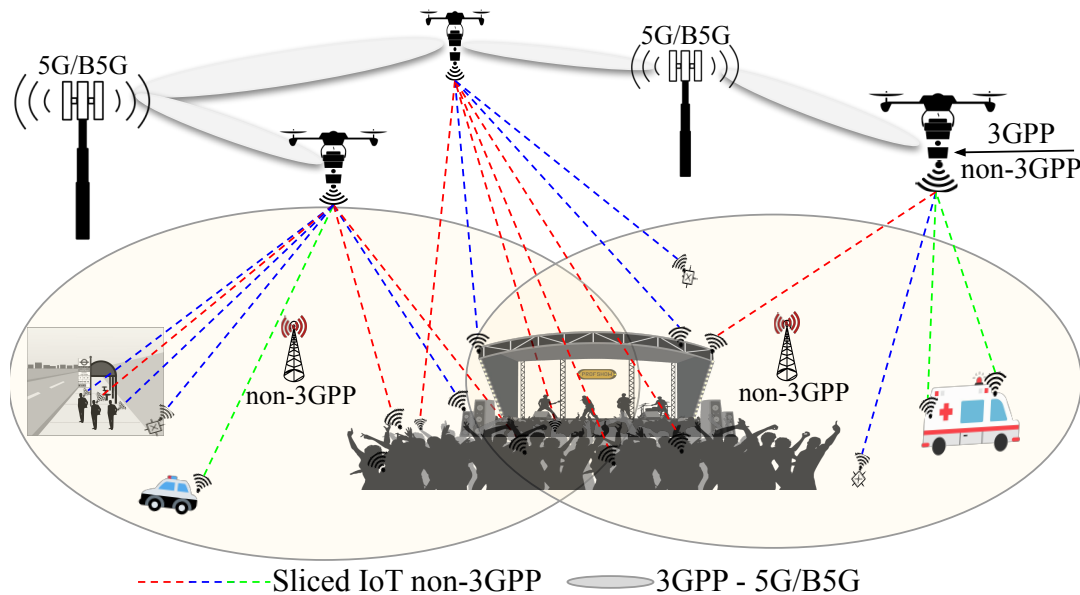


Figura 1.1: Cenário para o fatiamento de redes IoT não-3GPP envolvendo VANTs para suporte a surtos de demanda. [Desenvolvida pelo Autor.]

A Figura 1.1 ilustra um cenário de alocação dinâmica de recursos em um evento de grande porte, no qual a infraestrutura convencional torna-se insuficiente para atender à elevada demanda por conectividade. Para mitigar essa sobrecarga, a arquitetura proposta emprega **VANTs** equipados com *gateways* **LoRaWAN**, que são posicionados estrategicamente para conectar dispositivos **IoT** de maneira eficiente, assegurar a continuidade e manter a qualidade dos serviços. As linhas pontilhadas coloridas representam diferentes fatias lógicas da rede não-**3GPP**, cada uma dedicada a um tipo específico de serviço, e.g., monitoramento urbano, segurança pública e atendimento médico emergencial. Esse fatiamento permite o isolamento de tráfego e a alocação seletiva de recursos, adaptando-se às exigências de cada aplicação e garantindo o cumprimento dos acordos de nível de serviço (*Service Level Agreements (SLAs)*). Assim, a abordagem viabiliza uma infraestrutura temporária, escalável e resiliente, capaz de suportar o crescimento súbito da demanda em ambientes críticos.

De maneira complementar, a Figura 1.2 apresenta exemplos que destacam a versatilidade da solução em diferentes situações. Em eventos de alta densidade populacional,

como shows musicais ou eventos esportivos, a infraestrutura móvel pode ser rapidamente reforçada para lidar com o grande volume de usuários simultâneos, assegurando a continuidade e a qualidade da conectividade. Já em situações emergenciais, como deslizamentos de encostas, enchentes ou incêndios florestais, o modelo permite o posicionamento dinâmico e autônomo de **VANTs**, garantindo conectividade crítica para equipes de resgate e dispositivos **IoT** distribuídos em áreas afetadas. Essas características reforçam a aplicabilidade da solução em cenários de alta complexidade e demandas heterogêneas.

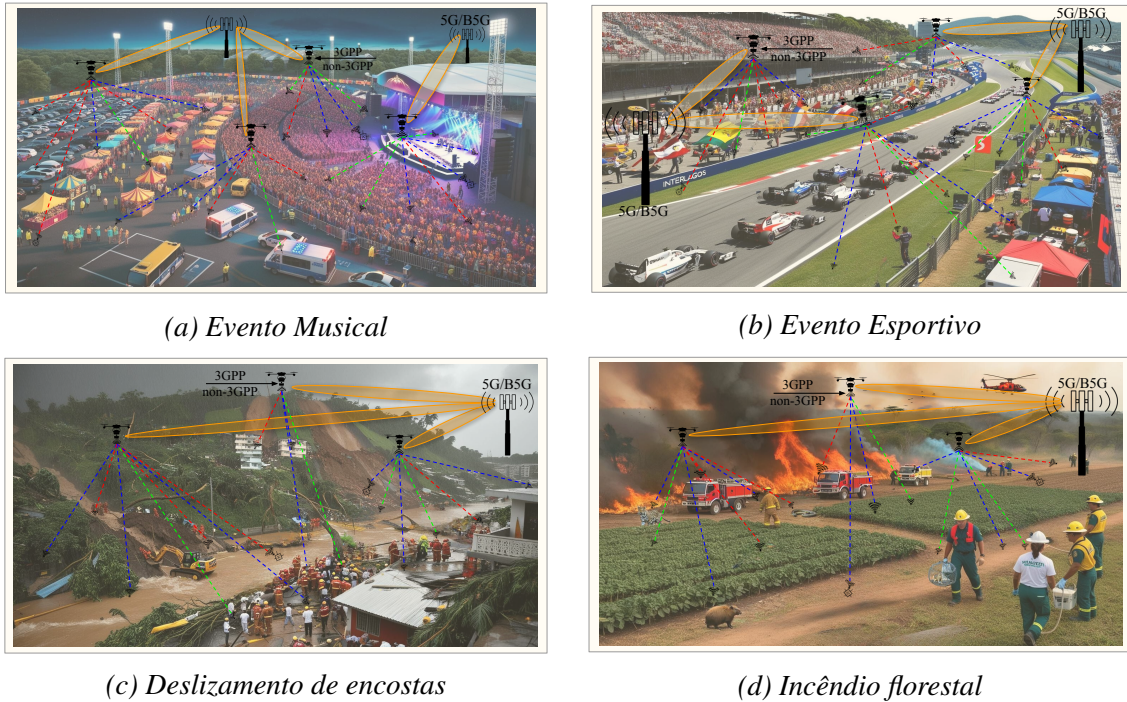


Figura 1.2: Exemplos de aplicação para suporte a surtos de demanda em diferentes cenários: eventos musicais, esportivos, deslizamentos de encostas e incêndios florestais. [Imagens geradas por IA com o uso do Canva Dream Lab, conforme descrito no Apêndice A.1]

A proposta desta tese é caracterizada por uma solução híbrida que integra métodos exatos e técnicas baseadas em *Machine Learning* (ML), especificamente **DRL**, para otimizar o posicionamento e o reposicionamento de **VANTs** em redes **IoT** não-**3GPP**. Essa abordagem híbrida garante a manutenção dos requisitos de **QoS** mesmo em cenários altamente complexos e dinâmicos, combinando robustez na fase inicial de alocação com flexibilidade na adaptação ao longo do tempo. Em resumo, esta pesquisa propõe não apenas uma solução técnica inovadora, mas também uma aplicação prática e versátil para redes **IoT** heterogêneas e resilientes. Ao responder aos desafios de conectividade em cenários dinâmicos e críticos, este trabalho contribui para a construção de redes de comunicação mais adaptáveis, eficientes e sustentáveis.

Trajetória da pesquisa

A trajetória desta pesquisa de doutorado teve início com os estudos preliminares sobre redes móveis de quinta geração (5G), com ênfase em conceitos de *softwarização*, na arquitetura baseada em serviços (*Service-Based Architecture (SBA)*), no núcleo da rede, e no acesso não-3GPP. Esses estudos iniciais possibilitaram a concepção de um minicurso apresentado em evento científico e a subsequente publicação de um capítulo de livro [Lobianco et al. 2021], consolidando os fundamentos teóricos que embasariam os avanços subsequentes da tese. A partir de então, aprofundaram-se as investigações em redes não-3GPP, o que culminou na realização do exame de qualificação e na delimitação inicial do escopo do trabalho.

Com o escopo definido, iniciaram-se os desenvolvimentos dos modelos baseados em MILP, que permitiram a formulação exata do problema de posicionamento dos VANTs e alocação de recursos. Esses modelos passaram por sucessivas melhorias, refletidas na produção de publicações científicas [Silva et al. 2023, Pires-Jr et al. 2024] que apresentaram os resultados e as contribuições obtidas com essa abordagem exata.

Posteriormente, a pesquisa avançou para o campo de ML, com o desenvolvimento de conjuntos de dados sintéticos por meio de simulações em ns-3, viabilizando os primeiros experimentos com aprendizado por reforço. Essa nova fase resultou na publicação de artigo científico sobre soluções com aprendizado profundo, consolidando o uso de DRL como abordagem complementar para decisões dinâmicas [Silva et al. 2024].

Ao longo do desenvolvimento da tese, as técnicas de DRL foram aprimoradas com a implementação e avaliação de diferentes algoritmos, incluindo as contribuições originais SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO, apresentadas na Seção 5.2. Tais avanços culminaram na elaboração do artigo submetido ao periódico *IEEE Access*, descrito no Item 4 da Tabela 6.1, que apresenta os resultados consolidados e as inovações desta pesquisa.

1.2 Questões de Pesquisa

A presente pesquisa busca avançar o estado da arte na alocação dinâmica de recursos em fatias de redes IoT não-3GPP, com foco em cenários críticos que exigem infraestrutura de comunicação resiliente, adaptável e escalável. Nesse contexto, são apresentadas três questões centrais que orientam as investigações realizadas no âmbito deste trabalho:

1. **Como prover conectividade eficiente, resiliente e com garantia de QoS em cenários de surtos de demanda por comunicação IoT não-3GPP?**

- **Hipótese:** Projetar e validar uma infraestrutura de comunicação emergencial e resiliente, baseada na implantação sob demanda de **VANTs** equipados com *gateways* **LoRaWAN**, capaz de manter os requisitos de **QoS** e atender aos acordos de nível de serviço (**SLAs**) mesmo em situações de sobrecarga ou falhas da infraestrutura fixa.
- **Impacto:** Garantia de continuidade e qualidade da conectividade em cenários críticos e de alta densidade populacional, por meio de uma solução escalável e adaptável que assegura desempenho estável frente a condições adversas e variações abruptas na demanda.

2. Como otimizar o custo e o desempenho na alocação de **VANTs** para suportar situações emergenciais?

- **Hipótese:** Desenvolver um modelo híbrido que combina **MILP** e **DRL** para otimizar a implantação dinâmica de **VANTs**, minimizando o número necessário de dispositivos e determinando suas posições ideais para atender às demandas emergenciais de comunicação. O **MILP** será usado para resolver o problema inicial de posicionamento e dimensionamento, enquanto o **DRL**, por meio dos algoritmos, **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**, ampliará os espaços de busca e melhorará a eficiência na tomada de decisões dinâmicas.
- **Impacto:** Redução significativa dos custos operacionais (**OPEX**) e de capital (**CAPEX**), ao implantar **VANTs** de forma estratégica, garantindo seu uso apenas onde e quando necessário. Essa abordagem promove a eficiência no uso de recursos, otimizando investimentos e operações. Além disso, a melhoria na infraestrutura da rede e a garantia de **QoS** elevam a qualidade e a confiabilidade dos serviços oferecidos. O uso de aprendizado por reforço profundo permitirá uma resposta mais adaptativa e eficiente às variações dinâmicas do ambiente e às demandas de comunicação, resultando em uma operação mais inteligente e sustentável.

3. Como atender a demandas heterogêneas por serviços **IoT** garantindo isolamento e priorização entre diferentes fatias de rede?

- **Hipótese:** Desenvolver um modelo de alocação dinâmica de recursos que organize e priorize as demandas em fatias de redes **IoT** não-**3GPP**, garantindo isolamento e eficiência no uso dos recursos.
- **Impacto:** Melhor aproveitamento dos recursos de rede, com garantia de **QoS** para aplicações críticas, como monitoramento em tempo real, enquanto mantém a eficiência em serviços menos exigentes.

Essas questões guiam o desenvolvimento de uma metodologia híbrida com soluções baseadas em técnicas de otimização e aprendizado por reforço profundo, bem como a integração de tecnologias emergentes. A proposta visa não apenas responder aos desafios atuais das redes IoT não-3GPP, mas também criar um ecossistema de comunicação robusto e adaptável, alinhado às exigências de conectividade em cenários críticos e de alta densidade.

1.3 Organização do trabalho

O presente trabalho está organizado conforme descrito a seguir:

- No Capítulo 2, o referencial teórico é apresentado. Ele envolve a definição dos conceitos basilares para o desenvolvimento desse trabalho. São descritos os fundamentos de redes LPWAN com ênfase em LoRaWAN, fatiamento de redes LoRaWAN, de VANTs e suas aplicações em redes, além dos fundamentos para MDP, Reinforcement Learning (RL) e DRL.
- O Capítulo 3 apresenta os avanços das tecnologias de comunicação e a forma que as redes heterogêneas têm impulsionado o desenvolvimento de soluções para otimização e gerenciamento de recursos em redes IoT. Além disso, apresenta-se a metodologia da pesquisa bibliográfica e são incluídas as contribuições recentes da literatura úteis para contextualizar os avanços dessa tese.
- O Capítulo 4 descreve a formulação do problema por meio de soluções exatas, baseadas em MILP para a alocação dinâmica de recursos em redes IoT com o uso de VANTs. São apresentadas uma revisão dos avanços recentes da pesquisa, a formulação do modelo proposto e as técnicas adotadas para garantir eficiência e escalabilidade na solução do problema, além das estratégias de avaliação e os resultados alcançados com os experimentos realizados.
- O Capítulo 5 apresenta a abordagem baseada em aprendizado de máquina para a alocação dinâmica de recursos em redes IoT com o uso de VANTs. Explora-se os modelos de aprendizado por reforço profundo e apresenta-se os algoritmos SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO com o objetivo de aprimorar a tomada de decisão na alocação dos recursos. Apresenta-se ainda as estratégias de avaliação do desempenho das soluções propostas e os resultados obtidos com os experimentos realizados.
- O Capítulo 6 apresenta a conclusão sobre as pesquisas e investigações aqui descritas, enumera as contribuições geradas como publicações, e as perspectivas e sugestões de trabalhos futuros.

- No Apêndice A, são apresentadas as ferramentas de Inteligência Artificial (IA) utilizadas no apoio ao desenvolvimento desse trabalho de tese e são incluídos exemplos de *prompts* utilizados.

Referencial teórico

Este capítulo apresenta a fundamentação teórica necessária para a compreensão do presente trabalho e culmina em um estudo da literatura no estado da arte, com o objetivo de posicionar esta proposta e estabelecer suas contribuições.

2.1 Redes não-3GPP, Fatiamento e Multi-RAT

A crescente demanda por conectividade em ambientes dinâmicos e heterogêneos impulsionou a integração de redes não-3GPP às arquiteturas de comunicação avançadas. Essas redes, que incluem tecnologias como Wi-Fi, Bluetooth, ZigBee, SigFox e, especialmente, LoRaWAN, não seguem as especificações do 3GPP, mas desempenham papel estratégico no suporte a aplicações IoT, devido ao seu baixo custo, consumo energético reduzido e capacidade de operar em faixas de frequência não licenciadas [3GPP 2021].

Para garantir interoperabilidade entre redes de diferentes naturezas, a arquitetura do núcleo 5G (5G Core (5GC)) adota o paradigma Multi-RAT, que permite a operação integrada de redes 3GPP e não-3GPP. Essa abordagem viabiliza a transição transparente entre múltiplas tecnologias de acesso, otimizando a cobertura e a eficiência energética, especialmente em cenários com dispositivos IoT de baixa potência e alta densidade [Mozny et al. 2020]. Em contextos urbanos e emergenciais, essa interoperabilidade é essencial para garantir continuidade de serviço e adaptação dinâmica às características do ambiente.

Complementarmente, o NS surge como mecanismo fundamental para o atendimento simultâneo de múltiplos requisitos de conectividade. Por meio do fatiamento, a infraestrutura física compartilhada pode ser particionada logicamente em fatias independentes, cada uma configurada para um perfil de serviço específico. Essa técnica assegura o isolamento do tráfego e permite a alocação seletiva de recursos de rede conforme as exigências de aplicações críticas, como sistemas de saúde, segurança pública, transporte inteligente e monitoramento ambiental [Popovski et al. 2018].

No contexto das redes não-3GPP, o NS permite que tecnologias como o LoRaWAN sejam configuradas com diferentes parâmetros operacionais — e.g., potência

de transmissão, fator de espalhamento (**SF**) e periodicidade de envio — criando múltiplas fatias adaptadas às necessidades de diferentes classes de dispositivos. Essa capacidade é particularmente vantajosa em cenários dinâmicos e de sobrecarga, nos quais **VANTs** podem ser utilizados como elementos móveis da infraestrutura, ajustando a topologia da rede em tempo real e garantindo a manutenção da **QoS** estabelecida nos acordos de **SLAs**.

Dessa forma, a integração entre redes não-**3GPP**, o suporte ao **Multi-RAT** e a aplicação de técnicas de fatiamento lógico formam os pilares da arquitetura proposta nesta tese. Esses elementos viabilizam a construção de soluções escaláveis, resilientes e adaptáveis, essenciais para a alocação dinâmica de recursos em redes **IoT** sob condições de demanda variável e infraestrutura limitada.

2.2 LoRa e LoRaWAN

As **LPWANs** são as redes de comunicação desenvolvidas para atender a demandas de acesso a longas distâncias, com baixa potência e baixo custo. No contexto das **LPWANs**, várias soluções são oferecidas no mercado de **IoT**, e.g., SigFox [SigFox 2021], NB-IoT [3GPP 2016] e **LoRaWAN** [LoRa Alliance 2015]. Entre essas alternativas **LoRaWAN** tem-se destacado, comparada às demais tecnologias, **LoRaWAN** possui boa capacidade para negociar com eficiência o alcance da comunicação em relação às altas taxas de dados, o que é importante para as aplicações **IoT** em escala urbana, e ainda, **LoRa** tem a capacidade de espalhar o sinal por uma banda de frequência mais ampla e é mais resiliente a interferências e a *jamming* [Daud et al. 2018].

Long Range Wide Area Network (**LoRaWAN**) é uma pilha de protocolos aberta que utiliza-se de **LoRa** como camada física. **LoRaWAN** define a arquitetura do sistema e o protocolo de comunicação da camada de acesso ao meio (*Media Access Control Layer* (**MAC**)), enquanto que **LoRa** é o protocolo da camada física (*Physical Layer* (**PHY**)) que habilita o enlace de comunicação de longo alcance. O protocolo e a arquitetura **LoRaWAN** têm a maior influência em determinar o tempo de vida de uma bateria de um dispositivo, a capacidade da rede, segurança, a qualidade de serviço e a grande variedade de aplicações oferecidas pela rede [LoRa Alliance 2020]. Além desses fatores, **LoRa** é agnóstica a implementações das camadas de nível mais alto, o que permite que ela interopere com outras arquiteturas de rede [Semtech 2015]. A Figura 2.1 apresenta as camadas do protocolo **LoRaWAN**.

Camada física, LoRa

Long Range (**LoRa**) é uma tecnologia de radiofrequência de longo alcance amplamente utilizada por dispositivos da **IoT**. Proposta pela [LoRa Alliance 2015] e desen-

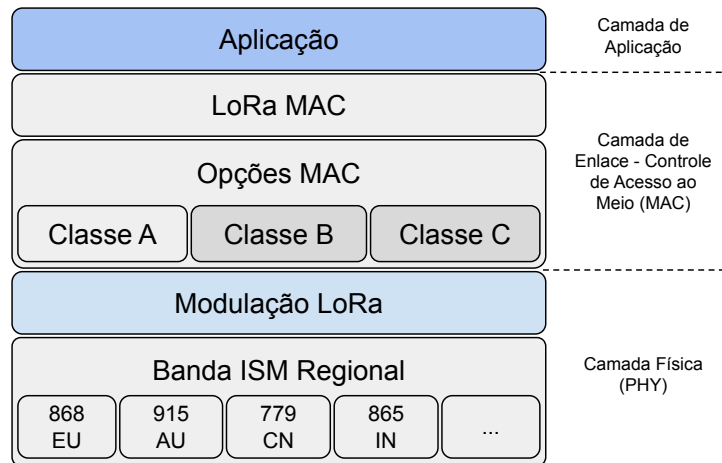


Figura 2.1: Protocolo LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [LoRa Alliance 2015].]

volvida pela [Semtech 2015]. LoRa é a camada física na modulação sem fio, a tecnologia é derivada da modulação *Chirp Spread Spectrum* (CSS) que mantém características de baixa potência, como nas modulações *Frequency Shifting Keying* (FSK), porém com um significativo incremento no alcance da comunicação. O CSS é usado há décadas em comunicações espaciais e militares devido às longas distâncias que o sinal pode alcançar, além da sua robustez à interferências. LoRa implementa uma taxa de transmissão de dados variável, utilizando fatores de espalhamento ortogonais, o que permite que o projetista do sistema troque a taxa de dados por alcance ou potência, de modo a otimizar o desempenho da rede em uma largura de banda constante. Essa otimização em LoRa é possível devido à capacidade de configuração de parâmetros, e.g., potência do rádio, sensibilidade, etc., o que permite modificar o comportamento dos dispositivos na rede [Semtech 2019].

Parâmetros LoRa

Os cinco principais parâmetros de configuração de um rádio LoRa são: largura de banda (*Bandwidth (BW)*), frequência da portadora (*Carrier Frequency (CF)*), taxa de codificação (*Code Rate (CR)*), fator de espalhamento (*Spreading Factor (SF)*) e potência de transmissão (*Transmission Power (TP)*).

BW é a largura das frequências na banda de transmissão. Uma **BW** mais alta fornece uma taxa de dados (*Data Rate (DR)*) mais alta, mas com uma sensibilidade mais baixa, devido à integração de ruído adicional. Por outro lado, uma **BW** mais baixa fornece uma sensibilidade mais alta, porém com uma taxa de dados mais baixa. Uma rede LoRa típica opera nas larguras de banda 125, 250 e 500kHz [LoRa Alliance 2015].

A **CF** é a frequência central, que pode ser programada em passos de 61Hz em valores que variam de 137MHz até 1.200MHz. O LoRa opera em bandas *Industrial, Scientific and Medical (ISM)*, e.g., 433MHz, 868MHz, etc., dependendo da região de

operação. No Brasil, a faixa de frequência disponível é de 915MHz até 928MHz, na Europa, estão disponíveis duas faixas: 433MHz e 863MHz até 868MHz, sendo a 868MHz a mais utilizada [Bor et al. 2016].

CR é a taxa de erro usada pelo modem **LoRa** para oferecer proteção contra rajadas de interferência. Essa taxa pode ser configurada conforme a relação entre dados reais e dados de erro, e.g., $CR = \frac{4}{4+c}$, onde $c = \{0, 1, 2, 3, 4\}$ representa a taxa de codificação. A relação $\frac{4}{4+0}$ indica ausência de proteção, enquanto $\frac{4}{5}$, $\frac{4}{6}$, $\frac{4}{7}$ e $\frac{4}{8}$ indicam níveis crescentes de proteção. Uma **CR** mais alta oferece maior proteção, mas aumenta o tempo no ar [Bor et al. 2016, Jez et al. 2021, Semtech 2019].

O **SF** define a relação entre a taxa de símbolos e a taxa de *chip*. Valores mais altos de **SF** permitem que o sinal viaje distâncias maiores e ainda seja recebido sem erros. No entanto, cada aumento de um no **SF** reduz pela metade a taxa de transmissão e dobra a duração da transmissão, aumentando o consumo de energia. Pacotes que usam diferentes **SFs** são ortogonais, o que significa que são invisíveis uns aos outros e aparecem apenas como ruído [Semtech 2019].

A **TP** em um rádio **LoRa** pode ser ajustada de -4 a $20dBm$, em passos de $1dB$. No entanto, devido às limitações de hardware, o intervalo prático geralmente está entre 2 e $20dBm$ [LoRa Alliance 2015].

Chirps, símbolos e chips

Na camada física **LoRa**, os dados são codificados na forma de *chirps*, símbolos e chips. *Chirps* são os sinais cuja frequência varia linearmente com o tempo dentro da largura de banda disponível, um *chirp* é considerado *up-chirp* se este varia de uma frequência mais baixa para uma mais alta, e *down-chirp* caso contrário. Um símbolo representa um conjunto de bits de dados, ou *chips*, o número de bits de dados modulados em um *chip* depende do **SF**, e.g. nove bits podem ser modulados em um *chip* com o **SF9**. O **SF** ainda define o número de *chips* transportados por um símbolo, e.g., para **SF12** são transmitidos $2^{12} = 4096$ *chips* por símbolo [Sundaram, Du e Zhao 2020].

Em **LoRa**, a taxa de *chirp* R_c depende apenas da largura de banda, ou seja, a taxa de *chirp* é igual a um *chirp* por segundo por Hertz de **BW**, e.g., $BW = 125$ kHz $R_c = 125000$ *chips/s*. A taxa de símbolo R_s é calculada por $R_s = BW/2^{SF}$, e.g., para $BW = 125$ kHz e $SF = 7$, $R_s = 125000/2^7 = 977$ símbolos/s. Para calcular a **DR**, **LoRa** inclui a **BW**, o **SF** e o código de correção de erros **CR**, A **DR** é calculada por

$$DR = SF \times \frac{BW}{2^{SF}} \times CR \text{ bits/s} \quad (2-1)$$

desta forma, ao aumentar o valor do **SF**, a **DR** diminui. A Tabela 2.1 apresenta as taxas de transmissão **LoRa** obtidas através da Equação 2-1, levando em consideração os

parâmetros usuais de **SF**, **CR** e **BW**.

Tabela 2.1: *DR segundo valores usuais de SF, BW e CR. [Desenvolvida pelo Autor.]*

SF	DR sem CR (kbps)			DR com CR=4/5 (kbps)			DR com CR=4/8 (kbps)		
	125kHz	250kHz	500kHz	125kHz	250kHz	500kHz	125kHz	250kHz	500kHz
7	6,84	13,67	27,34	5,47	10,94	21,88	3,42	6,84	13,67
8	3,91	7,81	15,63	3,13	6,25	12,50	1,95	3,91	7,81
9	2,20	4,39	8,79	1,76	3,52	7,03	1,10	2,20	4,39
10	1,22	2,44	4,88	0,98	1,95	3,91	0,61	1,22	2,44
11	0,67	1,34	2,69	0,54	1,07	2,15	0,34	0,67	1,34
12	0,37	0,73	1,46	0,29	0,59	1,17	0,18	0,37	0,73

A duração de um chip T_c é obtida por $T_c = 1/BW$, e.g., para $BW = 125 \text{ kHz}$, $T_c = 1/125000 = 8 \text{ } \mu\text{s}$. A duração de um símbolo T_s é obtida por $T_s = 2^{SF}/BW$. A Tabela 2.2 apresenta alguns exemplos.

Tabela 2.2: *Duração de um símbolo em relação ao SF e BW. [Criada pelo Autor.]*

BW	SF	T_s
125 kHz	7	1024 ms
250 kHz	7	512 μs
500 kHz	7	256 μs
125 kHz	9	4096 ms
125 kHz	12	32768 ms

Um **SF** mais alto aumenta a sensibilidade e o alcance, mas também prolonga o tempo de transmissão (*Time-on-Air (ToA)*) de um pacote [Semtech 2019]. Por exemplo, um dispositivo final localizado próximo a um *gateway* deve transmitir dados com um **SF** baixo, pois demanda pouco orçamento de *link*, e, conseqüentemente, alcança uma taxa de dados maior. No entanto, um dispositivo final localizado a vários quilômetros de um *gateway* precisará transmitir com um **SF** muito maior. Esse **SF** mais alto fornece um aumento de ganho de processamento e uma maior sensibilidade de recepção, embora a taxa de dados seja, necessariamente, menor. A Figura 2.2 demonstra essas relações.

Camada de acesso ao meio, LoRaWAN

Redes **LoRaWAN** são regulamentadas para operar sobre as bandas de frequência não licenciadas em sub-GHz **ISM**. As regulamentações de uso das bandas **ISM** variam de acordo com cada região, que definem as configurações possíveis para os enlaces **LoRa**, e.g., larguras de banda de 125kHz e 250kHz para *uplink* e 500kHz para *downlink*, **SF** que varia de 7 até 12, **TP** que varia entre 0 e 20dBm, e o ciclo de trabalho (*Duty*

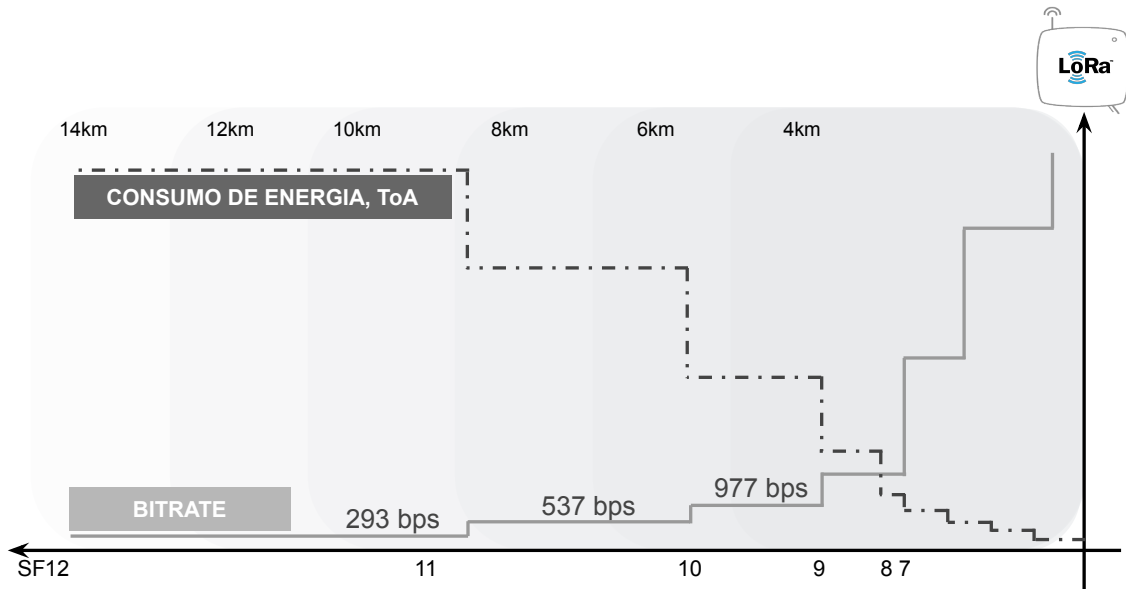


Figura 2.2: *ToA, taxa de dados e consumo energético em relação ao SF. [Desenvolvida pelo Autor. Adaptada de [Semtech 2019]]*

Cycle), que é limitado de acordo com os planos de frequência regionais, e.g., EU868-870, EU433, CN779-787 [LoRa Alliance 2021]. As limitações de ciclo de trabalho estão relacionadas com o controle do tempo em atividade de um dispositivo, e.g., se o ciclo de trabalho for determinado em 1%, significa que o dispositivo deverá esperar 100 vezes a duração do último quadro enviado até que ele possa enviar novamente no mesmo canal [Augustin et al. 2016].

Arquitetura LoRaWAN

A arquitetura **LoRaWAN** é composta pelos componentes apresentados na Figura 2.3.

- **Dispositivos LoRa: (LoRa End Devices (LoRa-ED))** são sensores de baixo consumo que usam **LoRa** para se comunicar com os *gateways* **LoRa**.
- **Gateways LoRa:** são dispositivos intermediários responsáveis pelo encaminhamento dos pacotes dos **LoRa-EDs** aos servidores de rede através de uma interface de *backhaul* IP. O *gateway* **LoRa** em uma rede **LoRaWAN** é um dispositivo simples e de baixo custo capaz de converter pacotes **LoRaWAN** em pacotes IP que podem ser enviados de através de um *backhaul* seguro ao servidor de rede **LoRa**.
- **Servidor de rede LoRa:** é responsável por detectar pacotes duplicados, decodificar os pacotes recebidos dos **LoRa-EDs** e gerar os pacotes que serão enviados de volta aos dispositivos.

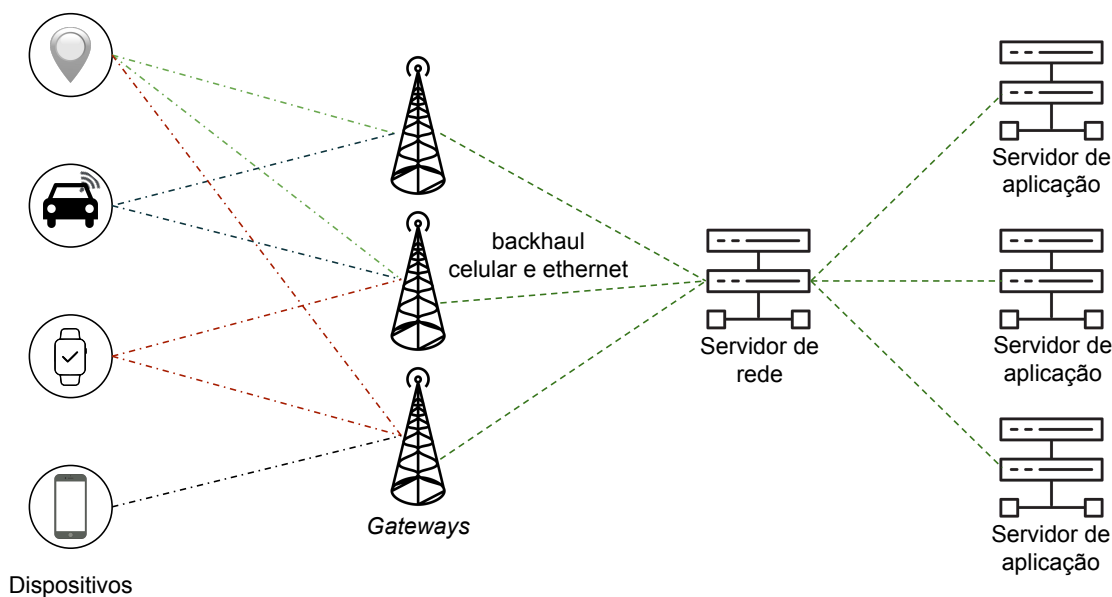


Figura 2.3: Componentes da arquitetura LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [Semtech 2019]]

- **Servidor de Aplicações LoRa:** é responsável por lidar com a camada de aplicação LoRaWAN, incluindo decodificação de dados de *uplink*, enfileiramento de *downlink* e codificação e criptografia de dados de *downlink*. Ele hospeda um servidor *Message Queuing Telemetry Transport (MQTT)* para *streaming* de dados de aplicativos, suporta *webhooks* HTTP, bem como integrações *pub/sub*.

Diferentemente das tradicionais redes móveis, em LoRaWAN os LoRa-ED não são associados a um *gateway* específico para acesso à rede, os *gateways* LoRa servem simplesmente como um relé da camada de enlace que encaminha os pacotes recebidos para servidores LoRaWAN após adicionar informações sobre a qualidade da recepção, ou seja, o LoRa-ED é associado a um servidor LoRaWAN que é responsável por detectar pacotes duplicados e selecionar um *gateway* LoRa apropriado para enviar uma resposta, caso seja necessário. Desta forma, os *gateways* LoRa são transparentes aos LoRa-EDs. O servidor LoRaWAN é responsável por definir as estratégias de reserva de recursos para o *gateway* LoRa e em configurar os LoRa-EDs com os parâmetros específicos de SF e TP [Dawaliby, Bradai e Pousset 2019].

Classe de dispositivos

LoRaWAN possui três classes de operação, ambas para comunicação bi-direcional. A classe A, modo básico, é suportada por todos os dispositivos, é o modo operacional mais comum. Nessa classe, os dispositivos decidem quando enviar dados, o que permite operar com consumo mínimo de energia. Após cada quadro enviado (*uplink*)

o dispositivo abre duas janelas de recebimento, liberando a recepção de dados dos *gateways* (*downlink*) e confirmações. Na classe B, em complemento às funcionalidades da classe A, os *gateways* podem transmitir quadros em intervalos regulares com agendamento da janela de recepção, e na classe C, os dispositivos podem receber quadros continuamente, exceto quando estão transmitindo. Todo dispositivo deve implementar a classe A. Aqueles que implementam mais de uma classe são considerados *higher Class End Device* (ED) [LoRa Alliance 2020].

Taxa de dados adaptativa

O protocolo LoRaWAN especifica o mecanismo de taxa de dados adaptativa (*Adaptive Data Rate* (ADR)). O ADR é uma tecnologia que simplifica os ajustes dos parâmetros de SF e TP com ênfase na melhoria da qualidade da comunicação. LoRaWAN possibilita escalar a rede pela simples adição de novos *gateways* e pode aumentar drasticamente a sua capacidade [Semtech 2019]. O mecanismo analisa as transmissões de *uplink* e de *downlink* e, após um conjunto de envios, a falta de uma transmissão de *downlink* possibilita detectar que houve alguma falha na comunicação. Ao detectar falha, o ADR tenta resolver o problema aumentando o TP até o seu limite superior. Caso o problema ainda não tenha sido resolvido, o procedimento é realizado aumentando o valor do parâmetro de SF e reajustando o TP para o mínimo novamente. Desta forma, as chances de que o enlace seja refeito são aumentadas [Kufakunesu, Hancke e Abu-Mahfouz 2020]. Entretanto, essa estratégia de adaptação do SF e da TP demanda muitos ciclos de envio e recebimento de pacotes pelo ED até perceber a necessidade de ajuste, e essa lenta convergência do algoritmo de ADR o torna vulnerável a transmissões simultâneas destrutivas, resultando em perdas por colisões massivas de pacotes e aumento no consumo de energia dos dispositivos [Li, Raza e Khan 2018, Lodhi, Wang e Farhad 2022]. São necessárias novas abordagens para a adaptação dos parâmetros LoRa que possibilitem melhorar a relação entre os parâmetros SF e TP com foco em maximizar transmissões bem-sucedidas e aproveitar ainda mais a característica de ortogonalidade, possibilitando, inclusive, a divisão dos recursos da rede LoRa em *slices*. O presente trabalho propõe fatiar a rede LoRaWAN através de uma heurística para realizar ajustes nos parâmetros da rede, além daqueles realizados sobre SF e TP, mas que também considera reposicionar os *gateways* para encontrar a melhor relação entre parâmetros, distância entre dispositivos, etc.

2.3 Network Slicing

Atualmente, grande parte das redes de computadores ainda é composta por uma variedade de dispositivos de hardware, monolíticos e proprietários, e conta com

configurações estáticas para o atendimento de suas demandas, o que torna ineficiente a operação e a introdução de novos recursos e serviços, devido ao aumento significativo dos CAPEX e OPEX [Chartsias et al. 2017, Zhang 2019]. Novas tecnologias e novos paradigmas de implantação e de operação de redes estão sendo desenvolvidos para lidar com essas questões.

O *5G System* (5GS) introduziu o conceito de rede flexível, programável e capaz de atender serviços com requisitos díspares e desafiadores, acelerando a transformação digital em diversos segmentos verticais [Barakabitze et al. 2020]. Um conceito disruptivo que pode realizar a visão do 5GS é o NS. Proposto pela *Next-Generation Mobile Networks* (NGMN) como um conceito fim-a-fim, o termo NS refere-se à criação e operação de múltiplas redes, logicamente independentes, sobre uma infraestrutura de telecomunicação física e compartilhada [NGMN Alliance 2016]. Segundo [3GPP 2018] o NS é a tecnologia que habilita o operador a criar redes personalizáveis para prover soluções otimizadas a diferentes cenários que demandam requisitos diversos, e.g., funcionalidade, desempenho e isolamento. Para o *International Telecommunication Union - Telecomm. Standardization Sector* (ITU-T) o NS é entendido como o particionamento em redes lógicas isoladas compostas de vários recursos virtuais, com plano de dados (*Data Plane* (DP)) e plano de controle (*Control Plane* (CP)) programáveis [ITU-T 2018]. O consenso dessas definições apresenta o NS como recurso para atender a demandas com requisitos distintos, mantendo isolamento e desempenho, de forma a permitir a criação de valor para segmentos verticais, provedores de aplicações e terceiros, permitindo uma operação de rede personalizada e uma diferenciação real de serviços.

O NS é construído sobre sete princípios fundamentais: automação, isolamento, customização, elasticidade, programabilidade, abstração hierárquica e a propriedade fim a fim [Afolabi et al. 2018].

- **Automação** habilita a configuração sob-demanda de NS sem necessitar de acordos fixos e intervenções manuais.
- **Isolamento** é a propriedade que assegura as garantias de desempenho e segurança para os inquilinos mesmo que usem serviços com requisitos conflitantes.
- **Customização** assegura que os recursos alocados sejam usados da forma mais eficiente para atender aos requisitos de serviço.
- **Elasticidade** é uma operação essencial aos recursos alocados para um *slice* em particular, de modo a assegurar o SLA desejado mesmo com variação das condições da rede.
- **Programabilidade** permite que terceiros controlem os recursos de fatia alocados, ou seja, recursos de rede e nuvem, facilitando a personalização sob-demanda dos serviços e a elasticidade de recursos.

- **Abstração hierárquica** é a propriedade em que o procedimento de abstração de recursos é repetido em um padrão hierárquico para todos os níveis para oferecer maior capacidade de abstração.
- **Fim a fim** é uma propriedade inerente ao fatiamento de rede que facilita a entrega do serviço dos provedores até o usuário final.

Arquitetura de *Network Slicing*

Para o **NS**, cada rede lógica, denominada *network slice* ou simplesmente *slice*, consiste em um conjunto de funções de rede virtualizadas (*Virtualized Network Functions (VNFs)*), bem como recursos de computação, armazenamento e rede para executá-las. Os *slices 5G* são altamente personalizáveis e flexíveis o suficiente para acomodar, simultaneamente, casos de uso diversos na mesma infraestrutura de rede [Zhang 2019]. Um modelo para *slices* (*Network Slice Templates (NSTs)*) é uma descrição completa da configuração, estrutura e fluxos de trabalho que definem como criar e controlar uma instância de *slice* durante seu ciclo de vida. Uma instância de *slice* (*Network Slice Instances (NSIs)*) é criada a partir de um **NSTs** e é composta por um conjunto de funções de rede (*Network Functions (NFs)*) gerenciadas em tempo de execução, e por recursos físicos, lógicos e virtuais para executar essas **NFs** [ITU-T 2012, NGMN Alliance 2016].

Uma **NSIs** pode ser implantada de forma estática ou dinâmica, **NSIs** estáticas são aquelas para as quais os recursos de rede são reservados previamente e que não sofrerão variações por mudanças no estado da rede durante o seu ciclo de vida. **NSIs** dinâmicas têm sua cota de recursos previamente reservados, porém os recursos vão sendo alocados dinamicamente para atender à variação temporal das demandas. A abordagem dinâmica é mais eficiente e flexível na utilização dos recursos da rede, mas é mais complexa para ser administrada, pois exige mecanismos adicionais para manter a **QoS**, enquanto a abordagem estática, apesar de mais simples, não é tão eficiente e pode acarretar o provisionamento excessivo de recursos [Raza et al. 2018, Zhou et al. 2020].

O ciclo de vida de uma **NSIs** é composto por quatro fases principais: preparação, instanciação, execução e descomissionamento. A Figura 2.4 apresenta as fases do ciclo de vida de um **NSIs**. Na fase de preparação, o *slice* ainda não existe, são realizados o pré-provisionamento dos recursos e a preparação do ambiente, ela inclui a criação e a verificação dos modelos de *slice*, a integração desses modelos, a preparação do ambiente de rede necessário para suportar o ciclo de vida das **NSIs** e quaisquer outras preparações que sejam necessárias na rede. Na fase de instanciação/configuração, todos os recursos compartilhados/dedicados ao **NSIs** são criados e configurados para um estado onde o **NSIs** está pronto para operação. A etapa de ativação inclui quaisquer ações que tornem o **NSIs** ativo, por exemplo, desviando o tráfego para ele, provisionando bancos de dados,

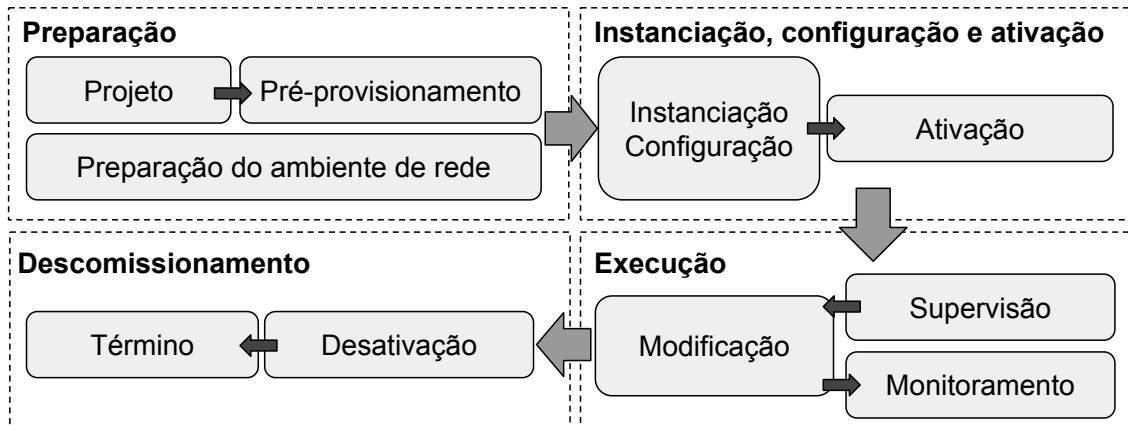


Figura 2.4: Ciclo de vida de um Slice. [Desenvolvida pelo Autor.
Adaptada de [3GPP 2020]]

etc., o *slice* é então criado, configurado e implantado sobre a infraestrutura do operador, de forma a atender aos requisitos especificados em seu descritor. Na fase de execução, o *slice* entra em operação e seu comportamento é monitorado para garantir o cumprimento dos requisitos. Finalmente, na fase de descomissionamento, o *slice* é encerrado e os recursos a ele atribuídos são liberados [3GPP 2020].

2.4 NS em LoRaWAN

O conceito de **NS** tem sido amplamente explorado em redes móveis, especialmente no contexto do **5G**, como uma abordagem para gerenciar eficientemente recursos heterogêneos e atender a diferentes requisitos de **QoS** em aplicações distintas. Recentemente, essa técnica tem sido adaptada para **LPWANs**, e.g., **LoRaWAN**, para lidar com os crescentes desafios de escalabilidade, heterogeneidade de dispositivos e requisitos específicos de aplicações **IoT**.

No caso de **LoRaWAN**, o **NS** permite a criação de múltiplas redes lógicas sobre uma infraestrutura física compartilhada, cada qual configurada para atender a diferentes necessidades de tráfego e **QoS**. A arquitetura padrão do **LoRaWAN**, que tradicionalmente utiliza o mecanismo **ADR** para ajustar **SF** e **TP**, tem sido expandida para suportar configurações baseadas em *slices* [Dawaliby et al. 2018]. Esses *slices* oferecem maior flexibilidade no gerenciamento de tráfego e permitem que cada dispositivo **IoT** seja associado a um *slice* com requisitos de latência, taxa de dados e confiabilidade bem definidos [Silva et al. 2023].

Uma abordagem central em **NS** para **LoRaWAN** envolve a reserva dinâmica de recursos por *slices*, utilizando métodos como a otimização baseada em estimativas de máxima verossimilhança (*Maximum Likelihood Estima-*

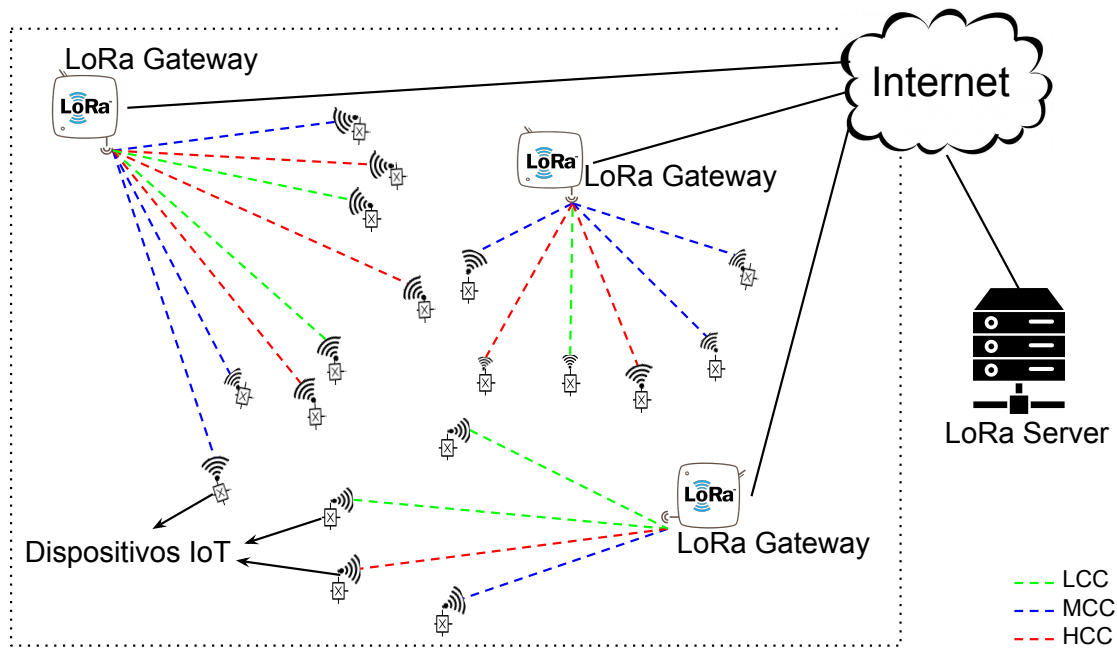


Figura 2.5: Arquitetura de network slicing em LoRaWAN. [Desenvolvida pelo Autor. Adaptada de [Dawaliby, Bradai e Pousset 2019]]

tion (MLE)) [Dawaliby, Bradai e Pousset 2019]. Esse processo busca evitar a saturação de canais, atribuindo recursos com base na prioridade dos *slices* e nos requisitos dos dispositivos associados. Além disso, algoritmos avançados, e.g., *TOPG* [Dawaliby, Bradai e Pousset 2021], têm sido propostos para otimizar a configuração de parâmetros de *SF* e *TP*, equilibrando requisitos de *QoS*, eficiência energética e confiabilidade dentro de cada *slice*.

A Figura 2.5 ilustra a implementação de *NS* em um ambiente LoRaWAN. Cada cor representa um *slice* lógico configurado para atender diferentes classes de *QoS*, e.g., baixa latência e alta prioridade, média latência e baixa prioridade. Nessa arquitetura, os dispositivos *IoT* comunicam-se com o *gateway* LoRaWAN, que então encaminha os dados para servidores LoRaWAN por meio da internet. Esse modelo demonstra como uma infraestrutura física pode ser segmentada logicamente para gerenciar as necessidades de tráfego e garantir isolamento eficiente entre os *slices*.

A segmentação apresentada na Figura 2.5 reforça os benefícios do *NS* ao permitir que dispositivos com diferentes demandas compartilhem a mesma infraestrutura sem comprometer o desempenho das aplicações críticas. Por exemplo, dispositivos atribuídos à *slices* de baixa latência e alta prioridade, como sensores de monitoramento em tempo real, recebem maior alocação de recursos e menor latência, enquanto aplicações menos críticas, como monitoramento ambiental, podem ser alocadas em *slices* de baixa prioridade. Essa abordagem também otimiza o uso de espectro, garantindo eficiência operacional mesmo em cenários de alta densidade de dispositivos.

Com a evolução das aplicações IoT, o uso de NS em LoRaWAN surge como uma solução promissora para atender à diversidade de demandas em cenários como cidades inteligentes, monitoramento ambiental e operações emergenciais. A integração de tecnologias emergentes, como VANTs equipados com gateways LoRaWAN, amplia ainda mais o potencial dessa abordagem, permitindo a criação de infraestruturas dinâmicas e adaptáveis que suportam surtos de demanda de comunicação e garantem a continuidade dos serviços mesmo em condições adversas.

2.5 Veículos Aéreos não Tripulados

VANTs são aeronaves pilotadas por controle remoto, ou por meio de programas de computador, sem um humano a bordo. Os primeiros VANTs da história surgiram com o objetivo militar e eram utilizados para ataques em territórios hostis [Zeng et al. 2020]. Nos dias atuais, uma série de aplicações para VANTs estão se popularizando. VANTs são empregados para, e.g., suporte a operações médicas [Ackerman e Koziol 2019, Ray e Nguyen 2020, Chen et al. 2021], logística e entregas [Troudi et al. 2017, Das et al. 2021], operações de busca e resgate [Ruetten et al. 2020, Schedl, Kurmi e Bimber 2021], inspeções de estruturas [George et al. 2019, Ribeiro et al. 2021], etc. Além dessas aplicações, várias pesquisas têm considerado o uso de VANTs como parte da infraestrutura de telecomunicações [Al-Turjman et al. 2019, Marchese, Moheddine e Patrone 2019, Jan, Hassan e Jung 2019, Khan et al. 2020, Parvaresh e Kantarci 2023, Moon et al. 2023].

A popularização do uso de VANTs e os riscos inerentes do aumento desses objetos no espaço aéreo têm direcionado os esforços das autoridades competentes para regulamentações do uso. No Brasil, o conjunto de normativas acerca dos VANTs foi definido pela Agência Nacional de Aviação Civil (ANAC) por meio do Regulamento Brasileiro da Aviação Civil Especial (RBAC-E) nº 94 [Brasil 2017]. Na Europa, a Comissão de Regulamentação da União Europeia lançou o Regulamento de Execução (RE) nº 2019/947, sobre as regras e procedimentos para a operação de aeronaves não tripuladas [Europe 2019]. A Tabela 2.3 apresenta uma síntese dessas normativas nos dois órgãos.

VANTs podem prover soluções para as comunicações sem-fio em um amplo conjunto de casos do mundo-real. Comparados às Base Stations (BSs) convencionais, os VANTs apresentam vantagens, e.g., ajuste de altitude, posicionamento dinâmico, estabelecimento de Line Of Sight (LoS) na comunicação com os dispositivos no solo, etc. [Mozaffari et al. 2016]. Uma importante aplicação para VANTs são os cenários da IoT, nos quais podem ser utilizados como retransmissores para melhorar a conectividade e a cobertura para EDs no solo [Shakhatreh e Khreishah 2018].

Tabela 2.3: Síntese das normas de utilização de VANTs no Brasil e na Europa, [Desenvolvida pelo Autor.]

Norma	Brasil	Europa
Legislação	Agência Nacional de Aviação Civil (ANAC) Regulamento Brasileiro da Aviação Civil Especial (RBAC-E) n° 94	Regulamento de Execução (RE) n° 2019/947 - Regras e procedimentos para a operação de aeronaves não tripuladas
Classes de operação	C1. Peso \geq 150kg C2. $25 \leq$ Peso \leq 150kg C3. Peso \leq 25kg	CA. Classe Aberta CE. Classe Específica CC. Classe Certificada
Altitude regulamentar (mín-máx)	30 - 120 metros	100 - 400 pés
Distância mínima (pessoas)	30 metros	100 pés
Registro	Obrigatório para C1 e C2	Obrigatório para CE e CC
Operação Autônoma	Proibida	Regulamentada por classe

2.6 Aprendizado de Máquina

O Aprendizado de Máquina (ML), uma subárea da IA, permite que algoritmos computacionais aprendam a partir de experiências passadas, aprimorando-se autonomamente [Sharma, Haque e Blaabjerg 2021]. Essa abordagem tem se consolidado como essencial para resolver problemas complexos e dinâmicos em diversos domínios, como redes IoT [Tellache et al. 2022], implantação de VANTs [Parvaresh e Kantarci 2023], NS [Toscano et al. 2019], etc. Ao capacitar sistemas a identificar padrões e tomar decisões inteligentes com base em dados, o MLs tem sido amplamente aplicado na otimização de recursos, previsão de comportamentos e adaptação em tempo real. Dentre suas subáreas, o Aprendizado por Reforço destaca-se por fornecer uma estrutura robusta para tomada de decisão em cenários de incerteza ou mudanças constantes, sendo ideal para aplicações em redes IoT heterogêneas e alocação dinâmica de recursos. Nesta seção, abordaremos os fundamentos do Aprendizado por Reforço (RL), sua modelagem matemática com o MDPs, e suas extensões, incluindo as redes neurais profundas (*Deep Neural Networks* (DNNs)) e algoritmos que ampliam sua aplicabilidade.

Aprendizado por Reforço

O RL baseia-se na interação contínua entre um agente e um ambiente para maximizar uma recompensa acumulada ao longo do tempo. Para formalizar essa dinâmica, utiliza-se a teoria dos MDPs, que fornece uma estrutura matemática para modelar problemas de decisão sequencial em ambientes estocásticos [Boucherie e Dijk 2017]. Esses

modelos permitem capturar de forma precisa como as ações de um agente influenciam o estado futuro do sistema e as recompensas associadas, estabelecendo a base teórica necessária para algoritmos de aprendizado por reforço. A seguir, exploraremos os conceitos fundamentais dos **MDPs** e sua relevância para o desenvolvimento de estratégias ótimas em sistemas dinâmicos.

Processos de Decisão de Markov

Os Processos de Decisão de Markov (**MDPs**) são modelos matemáticos amplamente utilizados para abordar problemas de tomada de decisão em sistemas dinâmicos sob incerteza. Fundamentados em uma estrutura estocástica, os **MDPs** são caracterizados por quatro elementos principais: um conjunto de estados (\mathcal{S}), um conjunto de ações (\mathcal{A}), uma função de transição de probabilidade ($\mathcal{T}(s'|s, a)$) e uma função de recompensa ($\mathcal{R}(s, a)$) [Hu e Yue 2008, Boucherie e Dijk 2017]. Esses elementos interagem para modelar a evolução do sistema ao longo do tempo e avaliar o impacto das ações tomadas pelo agente no ambiente, considerando a incerteza das transições [Pellegrini e Wainer 2007, Hu e Yue 2008]. A recompensa ($\mathcal{R}(s, a)$), em particular, mede a qualidade das ações executadas, incentivando comportamentos que maximizem os objetivos do sistema.

A Figura 2.6 ilustra a interação entre o agente e o ambiente em um **MDP**. O agente observa o estado atual, escolhe uma ação com base em uma política e o ambiente responde alterando o estado e fornecendo uma recompensa correspondente. Esse ciclo contínuo de ação, reação e aprendizado é essencial para sistemas dinâmicos que demandam adaptação em tempo real [Sutton e Barto 2018].

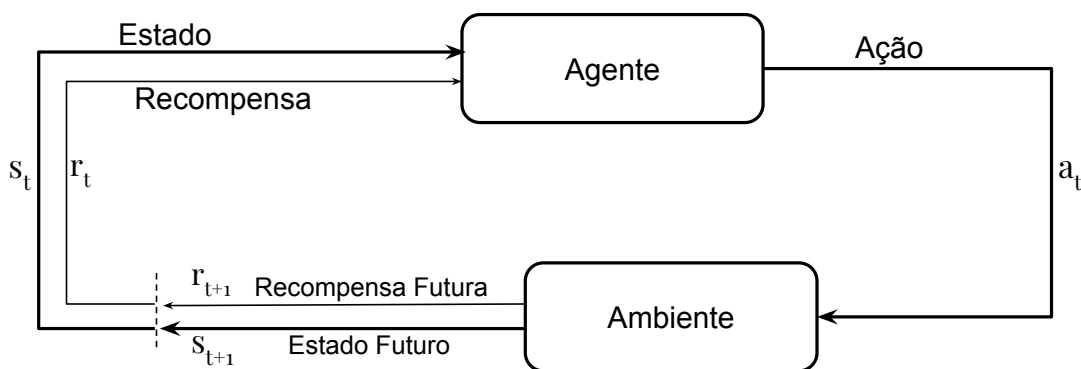


Figura 2.6: Interação entre o Agente e o Ambiente em um MDP. [Desenvolvida pelo Autor. Adaptada de [Sutton e Barto 2018]]

No contexto de redes **IoT**, os **MDPs** fornecem uma estrutura robusta para modelar decisões sequenciais em sistemas dinâmicos e incertos. Um estado pode representar,

por exemplo, a densidade de dispositivos conectados, enquanto as ações incluem ajustes de parâmetros como **TP** ou o reposicionamento de **VANTs**, visando melhorar a cobertura e atender aos requisitos de **QoS**. A recompensa reflete o impacto dessas decisões, avaliando métricas como redução de latência ou eficiência energética. Essa abordagem permite capturar com precisão as dinâmicas do sistema e otimizar recursos em cenários complexos e variáveis.

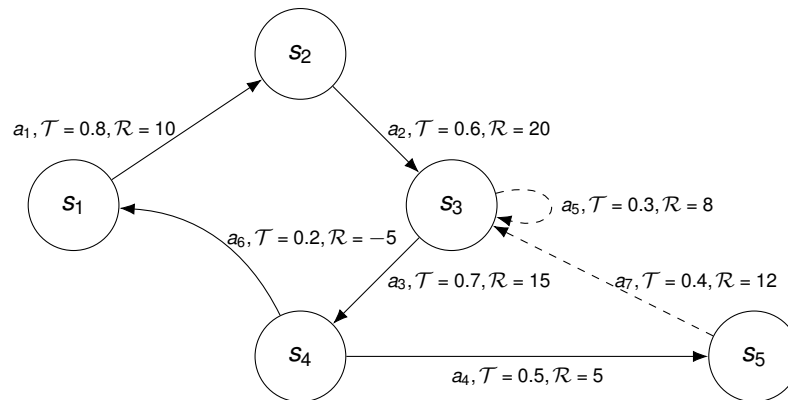


Figura 2.7: Diagrama de estados e transições para um MDP. Número de estados e transições arbitrários. [Diagrama produzido por IA conforme descrito no Apêndice A.2]

A Figura 2.7 apresenta um diagrama de estados e transições em um **MDP**, destacando as relações entre estados (S), ações (A), probabilidades de transição (T) e recompensas (R). Cada nó no diagrama representa um estado possível do sistema, enquanto as setas indicam as transições, acompanhadas por suas respectivas probabilidades e recompensas. Esse modelo visualiza a interação entre o agente e o ambiente, permitindo que o agente aprenda políticas ótimas ao explorar as dinâmicas do sistema e maximize recompensas acumuladas ao longo do tempo.

Os **MDPs** não apenas formalizam as dinâmicas de sistemas, mas também fornecem a base teórica para o aprendizado por reforço (**RL**). No **RL**, o agente aprende políticas diretamente das interações com o ambiente, sem a necessidade de um modelo explícito das transições. Essa evolução permite a aplicação eficiente em sistemas complexos e dinâmicos, como redes **IoT**, onde algoritmos como *Deep Q-Network* (**DQN**) e *Advantage Actor Critic* (**A2C**) viabilizam decisões em tempo real e alocação otimizada de recursos. A integração dos **MDPs** ao **RL** representa um avanço significativo na modelagem e otimização de ambientes altamente variáveis e heterogêneos.

Os **MDPs** fornecem a base teórica para modelar sistemas dinâmicos e incertos, o aprendizado por reforço expande esse conceito ao permitir que agentes aprendam políticas diretamente das interações com o ambiente. Um dos primeiros avanços nesse campo é o *Q-Learnings* (**QLs**), que se destaca por sua simplicidade e eficiência na busca por políticas ótimas.

Q-Learning

O *Q-Learning* (QL) é um dos algoritmos fundamentais de aprendizado por reforço, projetado para encontrar políticas ótimas em problemas modelados como MDPs [Sutton e Barto 2018]. Sua principal vantagem é a capacidade de aprender diretamente a partir da interação com o ambiente, sem depender de um modelo explícito das transições.

O QLs utiliza a função de valor $Q(s, a)$, que representa o valor esperado de executar uma ação a em um estado s . A atualização da função é dada por:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[\mathcal{R} + \gamma \max_{a'} Q(s', a') - Q(s, a) \right], \quad (2-2)$$

onde α é a taxa de aprendizado, \mathcal{R} é a recompensa imediata, γ é o fator de desconto, e $\max_{a'} Q(s', a')$ é a estimativa da melhor ação no próximo estado [Jang et al. 2019].

A aplicação do QLs enfrenta desafios significativos quando o espaço de estados e ações é muito grande, tornando inviável o uso de tabelas explícitas para armazenar os valores $Q(s, a)$. O algoritmo continua atualizando os valores da tabela Q iterativamente até convergir para uma política ótima π^* . Para superar as limitações de escalabilidade, técnicas baseadas em redes neurais têm sido amplamente empregadas. Essas redes permitem aproximar funções de valor de forma eficiente, expandindo o alcance do aprendizado por reforço para cenários complexos e de alta dimensionalidade, como os encontrados em redes IoT.

Redes Neurais

As redes neurais artificiais são estruturas computacionais inspiradas no cérebro humano, compostas por unidades interconectadas chamadas *neurônios*. Elas são utilizadas para modelar funções complexas, aproximando relações não lineares entre dados de entrada e saída [Zai e Brown 2020]. Uma rede neural (*Neural Network* (NN)) típica contém uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída, como ilustrado na Figura 2.8.

Cada camada processa os dados recebidos da anterior, aplicando uma função de ativação para determinar o comportamento não linear da rede. Esse mecanismo é essencial para resolver problemas de grande complexidade, e.g., no aprendizado por reforço [Rezazadeh 2024].

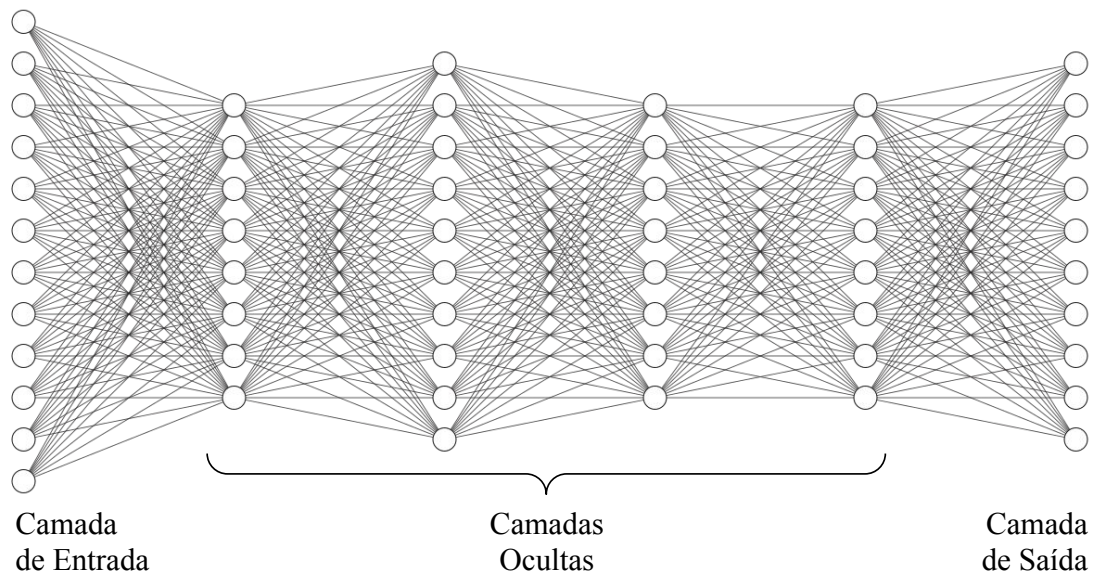


Figura 2.8: Estrutura típica de uma NN com camada de entrada, camadas ocultas e camada de saída. Número de nós e camadas arbitrários para melhor representação. [Criado pelo Autor. Diagrama produzido em: <http://alexlenail.me/NN-SVG/index.html>.]

Aprendizado Profundo

As redes neurais profundas, **DNNs**, representam uma evolução das redes neurais tradicionais. Elas utilizam múltiplas camadas ocultas para extrair, de forma hierárquica, características relevantes dos dados, tornando-se extremamente eficazes para resolver problemas de alta dimensionalidade e complexidade. Cada camada de uma **DNN** aprende representações progressivamente mais abstratas, permitindo que o modelo capture padrões complexos em dados brutos [Sutton e Barto 2018].

No contexto do aprendizado por reforço, as redes profundas desempenham um papel crucial ao aproximar funções de valor ou políticas que seriam impraticáveis de calcular explicitamente em ambientes com grandes espaços de estado-ação. Essa capacidade abriu caminho para algoritmos modernos que combinam aprendizado por reforço e aprendizado profundo, superando limitações de métodos anteriores, como o **QLs**. Essa integração deu origem ao campo conhecido como **DRL**.

Aprendizado por Reforço Profundo

O Aprendizado por Reforço Profundo (*Deep Reinforcement Learning* (**DRL**)) combina as técnicas de aprendizado profundo e aprendizado por reforço para enfrentar problemas em que os espaços de estados e ações são contínuos ou altamente dimensionais. Diferentemente de métodos como o **QLs**, que utilizam tabelas explícitas para armazenar

valores $Q(s, a)$, o **DRL** emprega redes neurais profundas para aproximar essas funções de valor, tornando viável sua aplicação em cenários mais complexos [Rezazadeh 2024].

A introdução de **DRL** trouxe avanços significativos, permitindo que agentes aprendessem a lidar com ambientes dinâmicos e incertos em áreas como robótica, jogos e redes de comunicação. Um dos primeiros marcos nesse campo foi o algoritmo *Deep Q-Network* (**DQN**), que utiliza redes neurais para melhorar o desempenho do **QLs**.

Deep Q-Networks

Nas *Deep Q-Networks* (**DQNs**), **DNNs** são utilizadas para aproximar a função $Q(s, a)$, eliminando a necessidade das tabelas explícitas do **QLs**. Essa abordagem permite que o agente lide eficientemente com espaços de estados contínuos e de alta dimensionalidade, onde as soluções tradicionais seriam inviáveis [Rezazadeh 2024]. A atualização dos parâmetros da **NN** é regida pela Equação 2-3,

$$\theta \leftarrow \theta + \alpha \nabla_{\theta} \left[r + \gamma \max_{a'} Q(s', a'; \theta') - Q(s, a; \theta) \right], \quad (2-3)$$

onde, $y = \mathcal{R} + \gamma \max_{a'} Q(s', a'; \theta^-)$ é o valor-alvo, θ representa os pesos da **NN** e θ^- são os pesos da rede-alvo, que são atualizados periodicamente para melhorar a estabilidade do treinamento, e α é a taxa de aprendizado [Zai e Brown 2020].

A função de custo utilizada para treinar a **NN** é baseada na Equação de Bellman, Equação 2-4, que estabelece uma relação recursiva entre os valores da função Q . Essa equação descreve a expectativa do valor futuro de uma ação considerando a soma da recompensa imediata e a melhor estimativa de valor do próximo estado. No contexto das **DQNs**, essa relação é formulada como um problema de minimização do erro quadrático médio (*Mean Square Error* (**MSE**)) entre a estimativa atual e o valor-alvo:

$$\mathcal{L}(\theta) = \mathbb{E}_{(s,a,\mathcal{R},s')} [(y - Q(s, a; \theta))^2], \quad (2-4)$$

e o valor-alvo y é definido como:

$$y = \mathcal{R} + \gamma \max_{a'} Q(s', a'; \theta^-), \quad (2-5)$$

onde, \mathcal{R} a recompensa obtida ao tomar a ação a no estado s , γ o fator de desconto, que pondera a importância de recompensas futuras, e θ^- os parâmetros da rede-alvo (*target network*). A rede-alvo é uma cópia estável da rede principal, usada para calcular y , e seus parâmetros são atualizados periodicamente para evitar oscilações no aprendizado [Sutton e Barto 2018].

A principal contribuição das **DQNs** é sua capacidade de generalização, permitindo que políticas aprendidas sejam aplicadas a estados previamente não observados.

Essa característica as torna adequadas para ambientes dinâmicos e complexos, como redes IoT ou robótica. No entanto, as DQNs são altamente sensíveis a hiperparâmetros, como a taxa de aprendizado (*learning rate*), a taxa de exploração (*exploration rate*) e a arquitetura da NN [Zai e Brown 2020]. Além disso, a necessidade de um ajuste cuidadoso desses parâmetros torna o design do modelo um aspecto crucial para o sucesso do algoritmo.

Apesar de sua eficácia, as DQNs enfrentam limitações em cenários onde as ações ou os estados são contínuos, pois precisam discretizar essas variáveis para aprendizado. Isso levou ao desenvolvimento de métodos alternativos e mais generalizáveis, como o framework *Actor Critic* (AC), que combina aprendizado baseado em política e aprendizado baseado em valor para lidar com problemas mais complexos.

Actor-Critic

O método *Actor Critic* (AC) é uma abordagem híbrida de aprendizado por reforço que combina as vantagens dos métodos baseados em políticas e dos métodos baseados em valor [Sutton e Barto 2018, Zai e Brown 2020]. Diferentemente de abordagens puramente baseadas em valor, como DQN, que aprendem uma função de valor $Q(s, a)$, o AC separa o aprendizado da política da avaliação das ações, tornando o treinamento mais eficiente e adequado para espaços de ações contínuos.

Essa técnica utiliza dois componentes principais, o ator (*actor*) aprende a política $\pi(a|s)$ diretamente, determinando quais ações devem ser tomadas com base no estado atual, e o crítico (*critic*) avalia as ações executadas pelo ator, estimando o valor do estado $V(s)$ e fornecendo um sinal de erro para guiar o aprimoramento da política. A atualização do ator ocorre com base no *feedback* do crítico, permitindo a melhoria gradual da política para maximizar a recompensa esperada. O ajuste da política segue a Equação 2-6

$$\theta_{\pi} \leftarrow \theta_{\pi} + \alpha \nabla_{\theta} \log \pi(a|s) \cdot \delta, \quad (2-6)$$

onde, θ_{π} representa os parâmetros da política (o ator), α é a taxa de aprendizado, que controla a magnitude da atualização, $\nabla_{\theta} \log \pi(a|s)$ indica o gradiente da política, ou seja, a direção em que a política $\pi(a|s)$ deve ser ajustada para melhorar o desempenho do agente, e δ é o erro de diferença temporal (*Temporal Difference* (TD)), que mede a discrepância entre a recompensa obtida e a estimativa do crítico. O TD é definido por:

$$\delta = r + \gamma V(s') - V(s), \quad (2-7)$$

onde, r é a recompensa imediata recebida ao executar a ação a no estado s , γ é o fator de desconto, que determina a importância das recompensas futuras, $V(s)$ é a estimativa do crítico para o valor do estado atual, e $V(s')$ é a estimativa do crítico para

o valor do próximo estado s' . Essa equação define a maneira como a política do ator é atualizada.

O gradiente da política ponderado pelo erro **TD** orienta a atualização na direção das ações que resultam em maiores recompensas, garantindo que a política do ator seja refinada ao longo do tempo com base no *feedback* do crítico. O erro **TD** mede a diferença entre a estimativa atual e a recompensa recebida somada ao valor estimado do próximo estado. Essa estratégia permite um aprendizado mais estável e eficiente do que métodos puramente baseados em política ou valor.

O método **AC** é particularmente eficaz em cenários contínuos e de alta dimensionalidade, onde o aprendizado baseado apenas em tabelas de valores se torna impraticável. Além disso, ao equilibrar a exploração de novas ações (*exploration*) com a exploração estratégica (*exploitation*) de ações promissoras, esse modelo melhora a eficiência da aprendizagem e acelera a convergência para políticas ótimas. No entanto, a qualidade do aprendizado depende fortemente da estabilidade do crítico e da correta definição da função de valor, o que pode tornar o treinamento desafiador.

Embora o método **AC** melhore a eficiência do aprendizado por reforço, ele ainda enfrenta desafios, como a alta variância nas estimativas de valor do crítico, o que pode dificultar a convergência da política. Para lidar com esse problema, foi introduzido o **A2C**, que aprimora a estratégia **AC** ao substituir a estimativa direta do valor do estado por uma métrica de vantagem (*advantage function*). Essa modificação reduz a variância do treinamento e melhora a estabilidade do aprendizado, tornando o processo de atualização mais eficiente.

Advantage Actor-Critic

O *Advantage Actor Critic* (**A2C**) é uma extensão do método **AC** que introduz o conceito de vantagem (*advantage*), o qual quantifica o impacto de uma ação em relação à média das ações possíveis no mesmo estado [Rezazadeh 2024, Zai e Brown 2020]. O principal objetivo do **A2C** é reduzir a variância no aprendizado, tornando as atualizações da política mais estáveis e eficientes. A função vantagem é definida como:

$$A(s, a) = Q(s, a) - V(s), \quad (2-8)$$

onde, $Q(s, a)$ representa a estimativa do retorno esperado ao executar a ação a no estado s , $V(s)$ é a estimativa do valor esperado do estado s , considerando todas as ações possíveis, e $A(s, a)$ mede o impacto da ação a ao comparar seu valor esperado com a média das ações no estado s . Dessa forma, se $A(s, a)$ for positivo, significa que a ação é melhor que a média e deve ser reforçada. Se for negativo, indica que a ação é pior que a média

e sua probabilidade de escolha deve ser reduzida. Isso permite que o modelo aprenda de forma mais eficiente, reduzindo a oscilação das atualizações da política.

A atualização do ator no **A2C** é ajustada para incorporar o termo de vantagem:

$$\theta_\pi \leftarrow \theta_\pi + \alpha \nabla_{\theta} \log \pi(a|s) \cdot A(s, a). \quad (2-9)$$

onde, θ_π são os parâmetros da política, α é a taxa de aprendizado, $\nabla_{\theta} \log \pi(a|s)$ é o gradiente da política, e $A(s, a)$ fornece um sinal de aprendizado mais estável, reduzindo a variância do treinamento.

O uso da função vantagem permite que o **A2C** melhore significativamente o desempenho em ambientes de alta dimensionalidade e com grande incerteza. No entanto, apesar dessas melhorias, a técnica ainda sofre com instabilidades no treinamento e dificuldades no ajuste dos hiperparâmetros. Além disso, grandes variações na atualização da política podem resultar em degradação do desempenho, caso as alterações sejam muito abruptas. Para solucionar esse problema, foi introduzido o *Proximal Policy Optimization* (**PPO**), um algoritmo que limita a variação excessiva na política durante o aprendizado, garantindo atualizações mais suaves e eficientes.

Proximal Policy Optimization

O *Proximal Policy Optimization* (**PPO**) é um algoritmo avançado de aprendizado por reforço baseado em políticas, projetado para melhorar a estabilidade e eficiência das atualizações da política [Schulman et al. 2017]. Ele surge como uma evolução do **AC** e sua variante **A2C**, abordando problemas de alta variância e instabilidade no treinamento.

Apesar das melhorias trazidas pelo **A2C**, esse método ainda sofre com oscilações abruptas na política aprendida, que podem levar à degradação do desempenho do agente. Essas oscilações ocorrem quando as atualizações são excessivamente grandes, desviando o agente de trajetórias de aprendizado eficientes. O **PPO** soluciona esse problema restringindo a magnitude das atualizações, garantindo mudanças mais suaves e controladas na política [Liu, Ding e Liu 2021].

O **PPO** introduz um mecanismo de restrição para evitar que a nova política π_θ se afaste excessivamente da política anterior $\pi_{\theta_{old}}$. Para isso, ele utiliza uma função de objetivo baseada na razão de probabilidade entre as políticas nova e antiga, definida como,

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}, \quad (2-10)$$

onde $\pi_\theta(a_t|s_t)$ representa a nova política parametrizada por θ , $\pi_{\theta_{old}}(a_t|s_t)$ representa a política antes da atualização, e $r_t(\theta)$ mede o quanto a nova política se desviou da antiga para a ação a_t . Com essa métrica, o **PPO** define sua função de otimização como,

$$\mathcal{L}_{PPO}(\theta) = \mathbb{E}_t [\min(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon)A_t)], \quad (2-11)$$

onde A_t é a função vantagem, que orienta a atualização da política, e ϵ é um hiperparâmetro que define o limite de variação da política. A função de *clipping* $\text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon)$ evita que a razão $r_t(\theta)$ ultrapasse a faixa $[1 - \epsilon, 1 + \epsilon]$, restringindo mudanças abruptas na política. Essa abordagem garante que as atualizações sejam feitas de forma controlada, reduzindo grandes oscilações e melhorando a estabilidade do aprendizado [Feriani e Hossain 2021].

O PPO possui duas principais variantes, a *PPO-Penalty*, que substitui a restrição rígida de atualização do *Trust Region Policy Optimization (TRPO)* por um termo de penalização no objetivo da política. Essa penalização atua diretamente na função de perda, evitando atualizações excessivas, e a *PPO-Clip*, que utiliza a técnica de *clipping* para limitar a magnitude das atualizações, sem a necessidade de um termo de penalização explícito. Essa variante se tornou a mais utilizada por sua simplicidade e eficiência computacional [Schulman et al. 2017]. O objetivo da função *clipping* do PPO é expressa como,

$$\mathcal{L}_{clip}(\theta) = \mathbb{E}_t [\min(r_t(\theta)A_t, g(\epsilon, A_t))], \quad (2-12)$$

onde,

$$g(\epsilon, A) = \begin{cases} (1 + \epsilon)A, & A \geq 0, \\ (1 - \epsilon)A, & A < 0. \end{cases} \quad (2-13)$$

Esse termo garante que a vantagem seja ajustada dentro de um intervalo de confiança, impedindo grandes oscilações. No contexto da abordagem Ator-Crítico, a função de perda total do PPO é definida como,

$$\mathcal{L}_{total}(\theta) = \mathcal{L}_{clip}(\theta) - K_1 L_V^F(\theta) + K_2 H(\pi_\theta), \quad (2-14)$$

onde, $L_V^F(\theta)$ é a perda da rede crítica (valor do estado), $H(\pi_\theta)$ representa o termo de entropia da política para incentivar a exploração, e K_1 e K_2 são coeficientes que controlam a influência de cada termo na otimização [Wang et al. 2023].

O PPO apresenta diversas vantagens em relação a métodos anteriores. Sua principal característica é a maior estabilidade no treinamento, pois a restrição na atualização da política reduz a variação excessiva entre iterações, garantindo uma convergência mais estável e previsível. Esse controle evita mudanças abruptas na política, o que poderia levar à degradação do desempenho do agente.

Além disso, o PPO oferece eficiência computacional, eliminando a necessidade de cálculos complexos de segunda ordem, como aqueles utilizados no TRPO. Essa

característica torna o PPO um algoritmo mais acessível e menos custoso em termos computacionais, permitindo sua aplicação em uma ampla gama de problemas.

Devido à sua robustez e estabilidade, o PPO tem sido amplamente aplicado em diferentes domínios. Em redes IoT, por exemplo, ele é utilizado para alocação dinâmica de recursos e otimização de parâmetros de rede. Sua capacidade de tomar decisões eficientes sem grandes variações de comportamento é essencial para garantir QoS em sistemas distribuídos e altamente variáveis [An e Wang 2024].

O PPO representa um avanço significativo no aprendizado por reforço baseado em políticas e continua sendo um dos algoritmos mais utilizados devido à sua eficiência e estabilidade. Sua abordagem de restrição na atualização da política permite um aprendizado mais confiável e consistente, tornando-o uma escolha ideal para aplicações de RL em larga escala.

2.7 Conclusão

Este capítulo apresentou os principais fundamentos teóricos necessários para o desenvolvimento desta pesquisa, abordando desde as tecnologias de redes emergentes, como 5G e LoRaWAN, até os conceitos avançados de NS e aprendizado por reforço profundo. A evolução das redes móveis e a integração de tecnologias LPWAN foram discutidas no contexto de suporte a aplicações de IoT, enfatizando a necessidade de mecanismos eficientes de alocação dinâmica de recursos.

A introdução do conceito de NS demonstrou seu papel fundamental na segmentação lógica de redes heterogêneas, possibilitando personalizar a QoS para diferentes classes de dispositivos e aplicações. Essa abordagem foi expandida para redes LoRaWAN, onde o particionamento da infraestrutura física em *slices* lógicos oferece uma solução promissora para os desafios de escalabilidade e gerenciamento de recursos. Além disso, a utilização de VANTs como elementos de infraestrutura dinâmica foi explorada, ressaltando sua aplicação na ampliação da cobertura de redes IoT e na otimização de conectividade em cenários dinâmicos.

No escopo do ML, foram investigadas tanto abordagens tradicionais quanto modernas de RL, com ênfase na fundamentação matemática baseada em MDPs para a modelagem de problemas sequenciais em ambientes estocásticos. A pesquisa evoluiu para o DRL, que integra a capacidade de generalização das redes neurais profundas com os mecanismos de aprendizado baseados em interação e recompensa do RL, viabilizando soluções mais escaláveis, adaptativas e eficientes. Diversos algoritmos de DRL foram analisados, e.g., DQN, *Double Deep Q-Network* (DDQN), A2C e PPO, destacando-se suas potencialidades e limitações na otimização de políticas de decisão sob condições de incerteza, alta dimensionalidade e variabilidade temporal dos estados do ambiente.

A revisão teórica realizada estabelece as bases para a formulação da metodologia proposta nesta pesquisa. A combinação de **NS**, **VANTs** e **DRL** configura um modelo inovador para a alocação dinâmica de recursos em redes **IoT não-3GPP**, permitindo adaptações inteligentes e eficientes às demandas de comunicação em tempo real. No próximo capítulo, será detalhada a abordagem metodológica empregada, com ênfase no desenvolvimento do modelo proposto e na avaliação de seu desempenho em cenários simulados. Com os fundamentos teóricos estabelecidos, o Capítulo 3, a seguir, apresentará os trabalhos relacionados, explorando abordagens existentes e destacando avanços e lacunas na área de alocação dinâmica de recursos em fatias de redes **IoT não-3GPP** envolvendo **VANTs**. Essa análise permitirá posicionar esta proposta no estado da arte, evidenciando suas contribuições e diferenciais.

Trabalhos Relacionados

Este capítulo aborda o avanço das tecnologias de comunicação e a crescente adoção de redes heterogêneas, que têm impulsionado o desenvolvimento de soluções inovadoras para otimização e gerenciamento de recursos em redes IoT. No contexto desta tese, destacam-se temas como *Network Slicing*, aprendido por reforço e otimização dinâmica de redes não-3GPP, aplicados a cenários que integram VANTs e redes IoT não-3GPP, como LoRaWAN. O objetivo é explorar trabalhos relacionados que abordam essas áreas, destacando as principais contribuições em métodos de alocação de recursos, ajustes de parâmetros de comunicação e estratégias para melhoria da QoS em sistemas complexos e dinâmicos. Foram considerados relevantes para esta análise os trabalhos que tratam de duas ou mais características essenciais ao escopo da pesquisa. Além disso, foram incluídas contribuições recentes, publicadas principalmente nos últimos cinco anos, a fim de capturar avanços contemporâneos na área. Trabalhos clássicos, que estabeleceram fundamentos teóricos ou metodológicos relevantes, também foram considerados para contextualizar os avanços recentes.

A metodologia utilizada para a pesquisa bibliográfica foi estruturada de forma a garantir a abrangência e a relevância dos trabalhos selecionados. O processo começou com a identificação de palavras-chave diretamente relacionadas às áreas de interesse, tais como *Network Slicing*, aprendido por reforço, otimização de recursos, LoRaWAN, VANTs, e redes 3GPP e não-3GPP. Essas palavras-chave foram combinadas para realizar buscas em bases de dados científicas amplamente reconhecidas, e.g., Periódicos CAPES, IEEE Xplore, SpringerLink e ScienceDirect, assegurando o acesso a artigos revisados por pares.

A análise dos trabalhos selecionados foi conduzida em etapas. Inicialmente, os resumos e introduções dos artigos foram avaliados para verificar sua relevância em relação ao tema da pesquisa. Posteriormente, os trabalhos elegíveis foram lidos integralmente, com atenção especial à descrição das metodologias, técnicas de solução e resultados obtidos. Essa abordagem permitiu identificar lacunas na literatura e posicionar a proposta desta tese em relação ao estado da arte.

Tabela 3.1: Características dos Trabalhos Relacionados. [Desenvolvida pelo Autor.]

	#	Artigo	Rede	NS	NT	VA	Técnica de Solução
Não-3GPP e NS	1	[Dawaliby, Bradai e Pousset 2019]	N3GPP	✓	✓		MLE
	2	[Dawaliby, Bradai e Pousset 2021]	N3GPP	✓	✓		MLE, TOPSIS, GMM
	3	[Messaoud et al. 2021]	N3GPP	✓	✓		MLE, MBGD
	4	[Messaoud, Bradai e Lagrange 2021]	N3GPP	✓	✓		DFQL
	5	[Messaoud, Bradai e Lagrange 2021]	N3GPP	✓	✓		OGMMC
	6	[Mardi et al. 2022]	N3GPP	✓	✓		K-means
	7	[Tellache et al. 2022]	N3GPP	✓	✓		DRL
	8	[Silva et al. 2023]	N3GPP	✓	✓	✓	MILP
VANTs	9	[He et al. 2018]	3GPP		✓	✓	PSO
	10	[Klaine et al. 2018]	3GPP		✓	✓	QL
	11	[Al-Turjman et al. 2019]	3GPP		✓	✓	GA, SA
	12	[Koushik, Hu e Kumar 2019]	SAGIN		✓	✓	DQN
	13	[Marchese, Moheddine e Patrone 2020]	N3GPP			✓	Testes em campo
	14	[Zhong et al. 2021]	3GPP		✓	✓	GA
	15	[Almeida, Campos e Ricardo 2022]	N3GPP		✓	✓	DDQN, DDPG
	16	[Lien e Deng 2022]	SAGIN		✓	✓	SR QD-learning
	17	[Su 2022]	3GPP		✓	✓	Otimização
	18	[Kirubakaran et al. 2023]	N3GPP		✓	✓	K-means, PSO, GA
	19	[Parvaresh e Kantarci 2023]	3GPP		✓	✓	ACDQL
	20	[Xu et al. 2023]	3GPP		✓	✓	PFDRl
	21	[Zhou et al. 2023]	SAGIN	✓	✓	✓	CDMADDPG
	22	[Gendia et al. 2024]	3GPP		✓	✓	MAB, DDQN
	23	[Lin et al. 2024]	3GPP		✓	✓	PLI
	24	[Silva et al. 2024]	N3GPP	✓	✓	✓	MILP-biObj
	25	[Song et al. 2024]	3GPP		✓	✓	MORL, TER
Este Trabalho			HETER	✓	✓	✓	Solução Híbrida: MILP-biObjetivo + DRL

SAGIN-Redes integradas Espaço, Ar e Terra, 3GPP-Redes 3GPP (4G/5G/6G), N3GPP- Redes Não-3GPP

HETER-Redes Heterogêneas, VA-VANTs, NS-Network Slicing , NT-Sintonia de parâmetros da rede

Para facilitar a análise comparativa dos trabalhos relacionados, os estudos selecionados foram organizados na Tabela 3.1, considerando suas características técnicas mais relevantes no contexto desta pesquisa. A estrutura da Tabela 3.1 está organizada para contemplar informações relevantes dos trabalhos relacionados, onde, **Rede**, indica o tipo predominante de infraestrutura analisada no estudo, distinguindo entre redes 3GPP, não-3GPP, e.g., LoRaWAN ou redes integradas, e.g., Space-Air-Ground Integrated Network (SAGIN). **NS**, demonstra se o trabalho aborda técnicas de Network Slicing para segmentação lógica da rede e atendimento a diferentes requisitos de QoS; **NT**, que representa a presença de mecanismos de network tuning, ou seja, ajustes finos de parâmetros de comunicação como SF, TP, largura de banda, etc. **VA**, que indica a utilização de VANTs como elementos ativos da arquitetura de rede, seja para cobertura, coleta de

dados ou atuação como *gateways*, e **Técnica de Solução**, que descreve a abordagem metodológica utilizada em cada estudo, incluindo algoritmos exatos, e.g., MILP, heurísticos, e.g., *Genetic Algorithms (GA)*, *Particle Swarm Optimization (PSO)*, *Simulated Annealing (SAn)*, métodos de aprendizado por reforço clássico, e.g., QL, ou profundo, DQN, A2C, PPO, *Deep Deterministic Policy Gradient (DDPG)*, etc.

Embora muitos trabalhos apresentem interseções entre os temas listados, cada estudo foi classificado na seção da revisão que melhor representa sua contribuição principal em relação ao escopo desta tese. Essa estruturação sistemática permite evidenciar os avanços obtidos pela literatura, as tendências metodológicas adotadas, bem como os aspectos que diferenciam e justificam a proposta desenvolvida neste trabalho. A análise da Tabela 3.1 apresentada nas Seções 3.1, 3.3, e 3.2 é também importante para a identificação de lacunas que motivam a formulação de uma abordagem integrada e multiobjetivo, apresentada nos capítulos subsequentes.

Este capítulo está organizado em três seções que agrupam os trabalhos relacionados ao tema. A Seção 3.1 explora os trabalhos com foco em redes não-3GPP, incluindo o fatiamento de redes não-3GPP. A Seção 3.2 aborda os trabalhos que tratam da utilização de VANTs como BSs ou *gateways* para redes IoT. A Seção 3.3 apresenta uma análise crítica dos trabalhos identificados na literatura, identifica as lacunas encontradas e apresenta as respectivas relações com os objetivos desta tese. A Seção 3.4 apresenta as considerações finais acerca do referencial bibliográfico, evidenciando contribuições e limitações gerais do referencial teórico para a proposta da tese.

3.1 Redes Não-3GPP e *Network Slicing*

Esta seção apresenta trabalhos que enfatizam o NS em redes não-3GPP. Apesar de alguns desses estudos tratarem de outros aspectos, como o uso de VANTs, sua principal contribuição reside no NS para a otimização de recursos.

O trabalho de [Dawaliby, Bradai e Pousset 2019] apresenta um modelo inovador para fatiamento dinâmico de redes IoT não-3GPP, especificamente LoRaWAN. O trabalho aborda os desafios de alocação de recursos e garantia de QoS em redes heterogêneas de IoT. Os autores propõem um algoritmo adaptativo baseado em MLE para alocação *inter-slices*, otimizando dinamicamente a reserva de recursos entre fatias de rede com base em requisitos de urgência e confiabilidade. Além disso, os autores apresentam um modelo de otimização *intra-slice* para organizar dispositivos e maximizar a eficiência da alocação de recursos, respeitando os requisitos específicos de QoS. Os resultados apresentados evidenciam melhorias significativas em isolamento entre fatias, confiabilidade de comunicação e eficiência energética. Este estudo representa uma contribuição essencial para o avanço de redes LoRaWAN. No trabalho [Dawaliby, Bradai e Pousset 2021]

os autores apresentam os avanços relacionados ao **NS** em redes **LoRaWAN** e descrevem um mecanismo inovador de fatiamento de rede em ambientes de cidades inteligentes, utilizando uma combinação de **MLE** e a técnica *Technique for Order of Preference by Similarity to Ideal Solution* (**TOPSIS**) integrada ao *Geometric Mean Method* (**GMM**). Os autores se concentram na otimização dinâmica dos parâmetros de **SF** e **TP** para lidar com desafios como interferência, eficiência energética e **QoS** em ambientes **IoT** heterogêneos. A abordagem apresentada visa garantir fatias virtuais isoladas com limiares específicos de **QoS**, segundo os acordos de nível de serviço (**SLA**), com foco em melhorar a confiabilidade da rede e a alocação de recursos, ao mesmo tempo que minimiza taxas de perda de pacotes e o consumo de energia. Simulações realizadas na plataforma **ns-3** validam a metodologia proposta, demonstrando melhorias significativas em termos de vazão, atraso e escalabilidade em implantações densas de **IoT**.

Seifeddine Messaoud et al. investigaram estratégias de fatiamento de redes voltadas para a *Industrial Internet of Things* (**IIoT**) em contextos dinâmicos e heterogêneos. No primeiro trabalho, eles avaliaram a eficiência de métodos de fatiamento baseados em **MLE** e *Mini-Batch Gradient Descent* (**MBGD**), considerando o impacto de configurações específicas de **LoRaWAN** em cenários industriais em larga escala. Os resultados mostraram que estratégias baseadas em predição **MBGD** superaram as alternativas em termos de respeito a limites de atraso, redução no consumo energético e aumento na vida útil das baterias dos dispositivos [Messaoud et al. 2021]. No segundo trabalho, foi proposto um modelo inovador de *Deep Federated Q-Learning* (**DFQL**), que combina aprendizado federado e aprendizado por reforço profundo para gerenciar alocação dinâmica de recursos em redes **IIoT**. A abordagem permite que agentes distribuídos otimizem localmente **SF** e **TP**, compartilhando experiências para construir um modelo global eficiente, que se mostrou superior em simulações [Messaoud, Bradai e Lagrange 2021]. No terceiro trabalho, os autores formularam um modelo multi-objetivo para alocar canais dinamicamente em **LoRaWAN**, considerando os requisitos de **QoS** e eficiência energética. Os autores propuseram o método de aprendizado *Online Gaussian Mixture Model for clustering* (**OGMMC**) para a associação dos dispositivos **IoT** aos *slices* **IIoT** mais apropriados. A solução explorou ainda a integração de arquiteturas baseadas em redes definidas por software (*Software Defined Network* (**SDN**)) e *Network Function Virtualization* (**NFV**), evidenciando avanços no desempenho da rede em termos de confiabilidade e eficiência [Messaoud, Bradai e Lagrange 2021].

[Mardi et al. 2022] apresentam um sistema eficiente de alocação para fatiamento centralizado em redes **LoRaWAN**, com foco em melhorar a confiabilidade da rede e garantir os **SLA** das fatias. O trabalho propõe um modelo de jogo coalicional, complementado por um algoritmo de *clustering K-means*, para particionar os nós **LoRa** em fatias iniciais com base em seus fatores de **SF**. Os jogadores, nós **LoRa**, podem migrar entre

coalizões para otimizar o desempenho das fatias em termos de taxa de sucesso de pacotes e consumo de energia da rede. Os resultados de simulação, conduzidos no simulador *LoRaSim*, demonstraram que o método proposto melhora significativamente a taxa de sucesso de pacotes e reduz o consumo energético, enquanto prioriza fatias com maior nível de prioridade. Adicionalmente, os autores mostram que a divisão dinâmica dos nós entre fatias é crucial para atender a serviços com diferentes requisitos de qualidade, abrindo caminho para aplicações **IoT** escaláveis e com maior eficiência energética.

O trabalho de [Tellache et al. 2022] apresenta uma abordagem baseada em **DRL** para alocação de recursos *intra-slice* em redes densas **LoRaWAN** utilizando o conceito de **NS**. A solução proposta substitui o mecanismo convencional de **ADR** por um esquema multi-agente baseado em **DQN**, no qual cada *slice* utiliza funções de recompensa específicas para atender aos requisitos de **QoS**. Os resultados experimentais demonstraram que a solução baseada em **DRL** supera o **ADR** em todas as métricas avaliadas, incluindo urgência, confiabilidade e eficiência energética, adaptando-se melhor às necessidades das fatias de rede e à densidade dos nós conectados. Este estudo ressalta o potencial do **DRL** para resolver problemas de decisão sequencial em ambientes dinâmicos e densos.

O trabalho de Silva et al. (2023) aborda a alocação dinâmica de recursos em redes **IoT** não-3GPP utilizando **VANTs** como *gateways* **LoRaWAN**. A pesquisa propõe um modelo de otimização **MILP** com o objetivo de minimizar o número de **VANTs** necessários para atender à demanda emergente, garantindo os níveis de **QoS** exigidos pelos dispositivos **IoT**. O modelo considera a flexibilidade de posicionamento dos **VANTs** e a configuração dos parâmetros de **SF** e **TP** para otimizar a comunicação. Resultados de simulação, utilizando o **ns-3**, demonstram que a abordagem proposta reduz significativamente o número de *gateways* implantados, mantendo um equilíbrio entre atraso e taxa de dados, em comparação com métodos de referência. Além disso, o estudo destaca a relevância do **NS** como ferramenta para alocação eficiente de recursos em cenários dinâmicos e críticos.

3.2 VANTs como parte da infraestrutura de comunicação

Nesta seção, são analisados trabalhos que exploram o uso de **VANTs** como **BSs** ou *gateways* para redes **IoT**. Embora algumas dessas propostas também considerem aspectos como **NS** ou aprendizado por reforço, sua contribuição principal está na utilização de **VANTs** para expansão e otimização de redes.

[He et al. 2018] propõem uma solução inovadora para o desafio de implantação de **BS** utilizando **VANTs** em terrenos irregulares. O trabalho aborda dois modelos

principais: o modelo de cobertura otimizada e o modelo de conectividade otimizada. No primeiro, os autores utilizam o algoritmo de PSO para determinar as posições tridimensionais dos VANTs, maximizando a QoS enquanto consideram as restrições de capacidade dos VANTs. Já no modelo de conectividade, o foco está em minimizar o número de VANTs necessários para conectar redes locais formadas por usuários através de comunicação ad hoc ou *Device to Device* (D2D). Os resultados das simulações demonstram que os métodos propostos não apenas melhoram a qualidade do canal ar-ar, mas também reduzem significativamente o número de VANTs necessários em diferentes cenários, como áreas urbanas densas e terrenos suburbanos, destacando a aplicabilidade prática da abordagem em cenários de comunicação emergencial e conectividade em terrenos complexos.

O trabalho de [Klaine et al. 2018] aborda o desafio de estabelecer redes de comunicação de emergência em cenários pós-desastres naturais, onde a infraestrutura convencional é frequentemente destruída. Os autores propõem uma solução baseada em aprendizado por reforço, especificamente QLS, para posicionar pequenas BSs transportadas por VANTs, denominadas *Drone Small Cells* (DSCs), em ambientes urbanos. A proposta visa maximizar a cobertura da rede e minimizar o número de usuários sem conexão, enquanto considera limitações de *backhaul* e da *Radio Access Network* (RAN). O algoritmo distribui de forma inteligente os vários DSCs, adaptando-se dinamicamente às condições de mobilidade e requisitos dos usuários, como as diferentes demandas de vazão entre equipes de resgate e usuários comuns. Resultados mostram que a abordagem baseada em QLS supera significativamente estratégias de posicionamento fixo em termos de cobertura e eficiência da rede, reforçando o potencial de soluções inteligentes para redes de comunicação de emergência em cenários dinâmicos.

Al-Turjman et al. [Al-Turjman et al. 2019] propõem uma estratégia otimizada para o posicionamento de VANTs como BS em redes 5G, com o objetivo de maximizar a cobertura em regiões críticas. O trabalho modela o problema como uma equação de otimização linear e utiliza os algoritmos meta-heurísticos de SAN e GA para determinar o número mínimo de VANTs e suas localizações ideais. Além de reduzir custos e consumo energético, a solução considera parâmetros de QoS, como taxa de dados, latência e vazão. Os resultados das simulações demonstram que o GA é mais eficiente para áreas maiores, enquanto o SAN apresenta soluções mais rápidas em áreas menores. Os autores destacam a aplicabilidade da abordagem em cenários como desastres naturais e áreas urbanas densas, sugerindo, ainda, estudos futuros para ambientes internos e VANTs dinâmicos.

O artigo de [Koushik, Hu e Kumar 2019] apresenta um modelo inovador baseado em aprendizado profundo para otimização do posicionamento de VANTs em uma topologia no formato de enxame para atendimento de redes dinâmicas. A proposta aborda o desafio de compensar enlaces rompidos em redes de comunicação de enxames de VANTs ao posicionar nós de retransmissão para atender aos requisitos de QoS. A abordagem

combina um modelo de fila *multi-hop* com prioridade (*Multi-hop Queuing Model with M/G/1 Preemptive Repeat Priority (MHQ-PRP)*) para analisar atrasos e taxas de entrega de pacotes com um algoritmo de **DQN** que otimiza a seleção de enlaces e posicionamento dos nós de retransmissão. O modelo integra parâmetros de múltiplas camadas de protocolo, incluindo *Signal-to-Interference-plus-Noise Ratio (SINR)*, *Packet Drop Rate (PDR)* e condições de interferência externa, proporcionando decisões baseadas em aprendizado profundo para alcançar a comunicação ótima. Resultados de simulação validam a eficiência do esquema proposto em aumentar a taxa de transferência de rede em comparação com métodos tradicionais, destacando sua aplicabilidade em redes de **VANTs** com topologias dinâmicas.

O trabalho de [Marchese, Moheddine e Patrone 2020] apresenta uma abordagem inovadora para a integração de **VANTs** como *gateways* para redes **IoT**, utilizando a tecnologia **LoRaWAN** e conectividade via satélites para estender a cobertura em cenários onde as redes terrestres são insuficientes ou inexistentes. O estudo se concentra em avaliar a viabilidade dessa solução, utilizando um cenário de agricultura inteligente como referência, com o objetivo de monitorar dados ambientais de sensores distribuídos. O artigo descreve o desenvolvimento de um sistema de teste que incorpora um **VANT** equipado com um *gateway IoT*, realizando experimentos para medir parâmetros como *Received Signal Strength Indicator (RSSI)*, *Signal-to-Noise Ratio (SNR)* e consumo de energia, tanto em campo quanto em laboratório. Os resultados destacam a viabilidade técnica da solução e a capacidade de transmitir dados para a plataforma em nuvem, mas apontam desafios, como o alto consumo de energia e as limitações impostas pelas características do ambiente externo. A proposta reforça a potencialidade dos **VANTs** como elementos-chave em redes **IoT** híbridas, especialmente para aplicações em áreas remotas, e sugere melhorias futuras, como o uso de algoritmos para otimização da cobertura e consumo energético.

O trabalho de [Zhong et al. 2021] propõe uma abordagem de otimização tridimensional para a implantação de **BS** em **VANTs**, com foco na maximização da cobertura dos usuários terrestres, enquanto considera limitações de capacidade e requisitos de **QoS**. O estudo destaca uma formulação que modela a relação entre perda de trajetória ar-solo e a localização dos **VANTs**, separando o problema em dois subproblemas: a determinação das posições horizontais em 2D e das altitudes em 3D. Utilizando um **GA**, os autores otimizam as localizações horizontais e altitudes dos **VANTs**, proporcionando maior eficiência energética e cobertura em comparação com abordagens tradicionais. Os resultados numéricos indicam que o método proposto supera significativamente técnicas como *K-means* e, especialmente, em cenários urbanos densos. Além disso, o trabalho demonstra robustez contra erros de localização dos usuários, tornando-o relevante para aplicações em ambientes dinâmicos e complexos, como cidades inteligentes e redes **IoT** emergentes.

O trabalho de [Almeida, Campos e Ricardo 2022] apresenta o algoritmo *Traffic-aware UAVs Placement Algorithm (TUPA)*, que utiliza aprendizado por reforço profundo para posicionar VANTs atuando como pontos de acesso voadores. O objetivo é otimizar a utilidade da rede em cenários com demandas de tráfego heterogêneas. A abordagem baseia-se em uma metodologia de aprendizado generalizável que permite ao TUPA adaptar-se a condições dinâmicas sem necessidade de treinamento adicional para cenários desconhecidos. Por meio de simulações no *framework ns3-gym*, o TUPA demonstrou melhorias significativas na utilidade da rede, superando soluções baseadas em maximização do sinal e abordagens de aprendizado específicas. A pesquisa contribui ao demonstrar que o treinamento com um conjunto diversificado de cenários aumenta a capacidade de generalização e destaca a eficácia de algoritmos como DA-DDDQN e DDPG em aplicações para redes de comunicação com VANTs. A relevância do estudo reside no avanço em alocação de recursos de redes aéreas em tempo real, suportando demandas emergentes de conectividade.

O trabalho de [Lien e Deng 2022] apresenta uma abordagem baseada em RL para a implantação autônoma de estações base não-terrestres (*Non-Terrestrial Base Station (NT-BS)*) em redes não-terrestres (*Non-Terrestrial Network (NTN)*). O estudo aborda os desafios únicos das NT-BSs móveis, como a necessidade de prever trajetórias de movimento de dispositivos usuários não-terrestres (*Non-Terrestrial User Equipments (NT-UEs)*) e outras NT-BSs, além de lidar com a variabilidade temporal na distribuição de NT-UEs. Para mitigar a alta complexidade computacional associada ao DRL, os autores propõem novos esquemas de RL distribuídos, denominados *k-step State Reduction Q-learning (SR-QLearning)* e *k-step State Reduction Distributive Q-learning (SR-DQLearning)*. Essas abordagens reduzem a dimensionalidade do espaço de estados e ações ao focar na taxa de variação de NT-UEs em diferentes localizações e introduzem decisões em intervalos otimizados de tempo. O estudo também realiza simulações que mostram que os métodos propostos superam esquemas convencionais, e.g., RL centralizado, maximização de SINR e DRL, em termos de número médio de NT-UEs atendidos.

O trabalho de [Su 2022] investiga a cobertura de redes sem fio de baixa altitude integradas a redes aéreas e terrestres no contexto de redes 5G. O autor propõe um modelo baseado na otimização de parâmetros, como TP e tilt da antena, adaptados a cenários específicos para melhorar a cobertura em baixas altitudes. A pesquisa inclui a análise e validação empírica dos efeitos dessas otimizações em ambientes reais, demonstrando melhorias no desempenho de parâmetros como *Synchronization Signal - Reference Signal Received Power (SS-RSRP)* e *Synchronization Signal - Signal-to-Interference-plus-Noise Ratio (SS-SINR)*. Essa abordagem apresenta contribuições significativas para otimizar redes integradas de baixa altitude, sendo essencial para cenários como gerenciamento de tráfego, resposta a emergências e proteção ambiental, onde a conectividade é crítica

para operações baseadas em VANTs. O trabalho também destaca as implicações para aplicações futuras de redes aéreas em contextos de alta demanda por conectividade.

[Kirubakaran et al. 2023] abordam soluções otimizadas de conectividade urbana para redes IoT, destacando o uso de *Unmanned Aerial Vehicle-mounted Base Stations* (UAV-BSs) em áreas densamente povoadas. O estudo combina algoritmos de aprendizado de máquina, como *K-means*, PSO e GA, para agrupar dispositivos IoT e determinar posicionamentos otimizados dos UAV-BSs, levando em conta as limitações de cobertura e capacidade de comunicação. Além disso, um método de agendamento em *mini-slots* de tempo foi implementado para melhorar a eficiência da transmissão de dados, reduzindo atrasos de comunicação e maximizando a taxa de dados agregada. A integração de BSs integradas em VANTs e conectadas por cabo às unidades robóticas *Cell-On-Wheels* (COW) demonstrou benefícios significativos em ambientes urbanos, reforçando a capacidade de cobertura e a robustez das conexões em cenários com alta densidade de dispositivos. Essa abordagem inovadora estabelece um modelo eficiente para redes IoT urbanas, utilizando tecnologias emergentes e métodos de otimização.

O artigo de [Parvareh e Kantarci 2023] explora o uso de BSs implantadas em VANTs (UAV-BSs) para superar as limitações das bases terrestres, principalmente em cenários de alta densidade urbana ou falhas na infraestrutura. A pesquisa propõe uma estratégia de aprendizado por reforço profundo baseada em *Actor-Critic Deep Q-learning* (ACDQL) para otimizar a localização tridimensional de UAV-BSs em ambientes dinâmicos com pontos terminais móveis. Resultados de simulação mostram que o modelo proposto aumenta significativamente a taxa de dados agregados da rede e reduz o tempo de convergência em comparação com modelos baseados em QLS e *Deep Q-Learning* (DQL). A solução adota a simulação combinada entre o ns-3 e agentes de aprendizado por reforço, o que evidencia sua aplicabilidade para cenários reais de comunicação em redes 5G e 6th *Generation Mobile Networks* (6G).

[Xu et al. 2023] apresentam uma abordagem inovadora para alocação conjunta de recursos e posicionamento de VANTs em redes 6G. A proposta central do trabalho consiste no uso de um algoritmo denominado *Personalized Federated Deep Reinforcement Learning* (PFDRL) para otimizar o vazão de longo prazo da rede, preservando a privacidade dos usuários e adaptando-se a estados dinâmicos da rede. O PFDRL combina modelos locais e personalizados treinados em VANTs com um esquema de agregação de parâmetros em dois níveis com um VANT líder, permitindo decisões em tempo real baseadas em observações locais e colaborativas. Resultados numéricos demonstram que o PFDRL supera soluções centralizadas e descentralizadas existentes em termos de convergência e vazão da rede, evidenciando sua eficiência para cenários com requisitos dinâmicos de QoS e mobilidade dos usuários.

O trabalho de [Zhou et al. 2023] propõe um modelo de otimização multiobje-

tivo para o fatiamento de redes em um ambiente integrado de espaço-ar-terra (SAGIN), utilizando três classes de *slices*, e.g., alta taxa de transmissão, baixa latência e ampla cobertura. A abordagem considera a alocação dinâmica de recursos físicos e virtuais, otimizando o posicionamento de VANTs virtuais, o compartilhamento *inter-slice* de sub-canais e potência, bem como a alocação *intra-slice* para usuários. O modelo é resolvido por meio de um algoritmo denominado *Central and Distributed Multi-Agent Deep Deterministic Policy Gradient* (CDMADDPG), que equilibra a complexidade computacional entre uma unidade centralizada e três unidades distribuídas, garantindo soluções próximas ao ótimo de Pareto. Resultados de simulação demonstram melhorias significativas na eficiência espectral e na QoS quando comparados a métodos tradicionais. Este trabalho destaca-se por integrar múltiplas camadas de rede e explorar algoritmos de otimização para cenários complexos e dinâmicos, oferecendo contribuições relevantes para o estado da arte em fatiamento de redes.

O artigo de [Gendia et al. 2024] propõe dois algoritmos baseados em RL para o planejamento eficiente de trajetórias de VANTs em redes de comunicação habilitadas por *Millimeter Wave* (mmWave) e utilizando *Non-Orthogonal Multiple Access* (NOMA). A pesquisa busca otimizar simultaneamente a trajetória dos VANTs, a seleção de dispositivos receptores e a alocação de TP, maximizando a taxa de soma ergódica¹ e preservando a energia dos VANTs em cenários dinâmicos e com canais sem fio desafiadores. A abordagem utiliza um modelo *Multi-armed Bandit* (MAB) e DDQNs para projetar ações eficientes, com testes indicando desempenho próximo do ótimo com significativas economias de energia. Os resultados demonstram a viabilidade de tais esquemas em redes emergenciais e aplicações para 5G e 6G.

O trabalho de [Lin et al. 2024] investiga redes SAGIN com suporte de VANTs e *caching*, propondo abordagens otimizadas para maximizar a taxa mínima de usuários. O estudo apresenta dois problemas de otimização, um que considera o *caching* pré-definido e outro que inclui a otimização conjunta de *caching*, comunicação e trajetórias de VANTs. A solução envolve a decomposição dos problemas em subproblemas resolvidos iterativamente por métodos como *Successive Convex Approximation* (SCA) e otimização linear inteira. Simulações demonstram que as abordagens propostas superam métodos de referência, evidenciando a eficiência na alocação de recursos e melhoria de desempenho em redes heterogêneas, especialmente em cenários de alta demanda por conectividade.

O trabalho de [Silva et al. 2024] aborda a integração de VANTs e redes IoT não-3GPP LoRaWAN com fatiamento de redes para melhorar a QoS em cenários IoT dinâmicos. A proposta inicial utiliza MILP para minimizar o número de VANTs necessários

¹Em redes de comunicação e no planejamento de trajetórias de VANTs, a **soma ergódica** é empregada para capturar o desempenho médio da taxa de transmissão de dados de dispositivos distribuídos em um ambiente dinâmico.

e otimizar suas posições, mas a complexidade do problema demonstrou ser NP-difícil. Como alternativa, os autores modelaram o problema como um MDP e aplicaram técnicas de DRL, especificamente os algoritmos DQN e A2C. Os resultados, obtidos por meio de simulações integradas ao ns-3, mostram que as soluções baseadas em DRL oferecem ganhos significativos em termos de QoS, superando métodos de otimização tradicionais. Além disso, as técnicas propostas demonstram viabilidade para cenários de alta escala, evidenciando sua robustez e aplicabilidade. A proposta é uma evolução de trabalhos anteriores, com melhorias no espaço de busca e na integração de restrições de QoS específicas para IoT não-3GPP.

O trabalho de [Song et al. 2024] apresenta uma abordagem de aprendizado por reforço multiobjetivo para a otimização da trajetória de VANTs em um cenário de *Multi-Access Edge Computing* (MEC) assistido por VANTs com carregamento sem fio. O estudo aborda os desafios de equilibrar objetivos conflitantes, como eficiência energética e coleta de tarefas computacionais, em um ambiente dinâmico. Para isso, os autores propõem um modelo *Multi-Objective Markov Decision Process* (MOMDP) e um algoritmo de aprendizado por reforço multiobjetivo com um esquema de repetição de experiência baseado em traços (*Multi-Objective Reinforcement Learning Trace-based Experience Replay* (MORL-TER)). Os resultados experimentais demonstram que a solução proposta supera abordagens de estado da arte, adaptando-se eficientemente às preferências dinâmicas e alcançando um equilíbrio favorável entre os objetivos conflitantes. Este trabalho contribui significativamente para o campo de redes assistidas por VANTs, ao propor um método inovador que combina técnicas de aprendizado avançadas com otimização de múltiplos objetivos.

3.3 Análise Crítica, Lacunas e Comparações

A revisão conduzida neste capítulo evidencia avanços significativos nas áreas de NS, VANTs e DRL aplicados a redes IoT heterogêneas. Os trabalhos analisados forneceram contribuições relevantes tanto na perspectiva da otimização de recursos quanto na expansão da cobertura por meio de soluções móveis. No entanto, a partir da análise dos trabalhos existentes e suas respectivas relações com os objetivos desta tese, observa-se a presença de lacunas que ainda limitam a aplicabilidade plena das abordagens na literatura.

Uma das principais limitações diz respeito ao escopo restrito da maioria das propostas. Diversos estudos concentram-se em tecnologias específicas, como redes LoRaWAN [Dawaliby, Bradai e Pousset 2019, Messaoud, Bradai e Lagrange 2021] ou redes 3GPP [Xu et al. 2023], sem considerar a integração com infraestruturas heterogêneas compostas por múltiplos domínios de rede. Isso compromete a generalização das

soluções em ambientes reais, onde o suporte simultâneo a redes 3GPP e não-3GPP é essencial para a operação de sistemas IoT em larga escala.

Adicionalmente, observa-se que grande parte dos trabalhos adota abordagens de otimização isoladas, sejam elas baseadas em heurísticas [Al-Turjman et al. 2019], modelos matemáticos exatos [Silva et al. 2023], ou algoritmos de DRL [Almeida, Campos e Ricardo 2022]. Poucos estudos exploram modelos híbridos que combinem a precisão de formulações exatas com a adaptabilidade das técnicas de RL, o que limita a escalabilidade e a capacidade de generalização frente à dinâmica dos cenários operacionais.

A aplicação de algoritmos de DRL também apresenta lacunas metodológicas. Muitos estudos ainda se apoiam em arquiteturas clássicas como DQN ou Q-Learning [Klaine et al. 2018], frequentemente validadas em ambientes simplificados, sem o uso de simuladores realistas como o ns-3. A ausência de testes em ambientes com modelagem física e tráfego realístico compromete a robustez das propostas frente a condições variáveis de canal, mobilidade e densidade de dispositivos.

Outra limitação importante é a falta de abordagens que considerem múltiplos objetivos simultaneamente. Problemas reais de redes IoT envolvem a otimização de métricas conflitantes, como eficiência energética, cobertura, latência e uso de espectro. Ainda que existam propostas recentes com foco multiobjetivo, como as de [Song et al. 2024, Zhou et al. 2023], estas não contemplam a integração com redes não-3GPP nem a presença de VANTs como elementos de mobilidade estratégica e reconfiguração da rede.

Frente a esse cenário, a proposta desta tese se diferencia ao integrar múltiplas dimensões tecnológicas, i.e., suporte a redes heterogêneas não-3GPP, uso de VANTs como elementos móveis e adaptativos, aplicação conjunta de MILP e DRL, formulação multiobjetivo, e validação em ambiente simulado de alta fidelidade. Tais aspectos posicionam a proposta como uma contribuição inédita e abrangente no estado da arte, apta a enfrentar os desafios atuais de conectividade em redes IoT críticas e dinâmicas.

3.4 Conclusão

Este capítulo apresentou uma revisão abrangente da literatura sobre fatiamento de redes, uso de VANTs como infraestrutura adaptativa, técnicas de alocação de recursos e algoritmos de aprendizado por reforço aplicados ao contexto de redes IoT. A análise crítica realizada revelou tanto os avanços obtidos quanto as limitações persistentes nos trabalhos atuais, destacando lacunas em termos de integração tecnológica, complexidade das abordagens, fidelidade de simulação e suporte a objetivos múltiplos.

As contribuições revisadas serviram como base teórica sólida para fundamentar as decisões metodológicas adotadas nesta tese, ao mesmo tempo em que reforçaram a

necessidade de uma abordagem inovadora e integrada. Ao propor um modelo híbrido baseado em **MILP** e **DRL**, com suporte a **NS** e conectividade não-**3GPP**, esta tese visa preencher as lacunas identificadas, contribuindo de forma original para o avanço do estado da arte em redes **IoT** com mobilidade e heterogeneidade.

Compreendidos os fundamentos teóricos e o contexto dos trabalhos relacionados, o Capítulo 4, será apresentada a primeira fase da proposta desta tese, que consiste na formulação de um modelo de otimização exata baseado em **MILP**, capaz de determinar o posicionamento ótimo de **VANTs** e realizar a alocação inicial de recursos entre as fatias de rede, respeitando os requisitos de **QoS** definidos para cada classe de serviço **IoT**.

Abordagem Exata

O Capítulo 4 apresenta uma abordagem exata, baseada em Programação Linear Inteira Mista (MILP), para a alocação dinâmica de recursos em redes IoT com o uso de VANTs. Neste capítulo, o leitor encontrará uma revisão dos avanços recentes nesta pesquisa, a formulação do modelo proposto e as técnicas adotadas para aumentar eficiência e escalabilidade na solução do problema. Além disso, são descritas as estratégias de avaliação e os resultados alcançados com os experimentos realizados.

4.1 Evolução da pesquisa com MILP

A pesquisa sobre alocação dinâmica de recursos em redes IoT com o uso de VANTs apresentou avanços significativos ao longo do desenvolvimento desta tese. Essa trajetória evolutiva pode ser observada a partir das contribuições publicadas durante o período de investigação [Silva et al. 2023, Pires-Jr et al. 2024, Silva et al. 2024], as quais compõem uma linha de pesquisa, voltada à proposição de soluções que conciliem eficiência operacional, escalabilidade e aderência aos requisitos de QoS em cenários de comunicação heterogêneos e complexos.

O trabalho seminal apresentado em [Silva et al. 2023] introduziu um modelo baseado em MILP com o objetivo de minimizar a quantidade de VANTs necessários em redes IoT não-3GPP, priorizando a otimização das posições dos VANTs e a associação dos dispositivos IoT a *slices*, considerando os parâmetros de SF e TP. Essa contribuição estabeleceu as fundações para a integração do NS com tecnologias LoRaWAN, propondo soluções que já indicavam avanços expressivos na mitigação de interferências e na garantia dos requisitos de QoS em ambientes de alta densidade.

Na sequência, nos trabalhos em coautoria com [Pires-Jr et al. 2024] e no estudo autoral [Silva et al. 2024], essa abordagem foi refinada com a ampliação do espaço de busca do modelo MILP e a incorporação de novas restrições voltadas a cenários mais desafiadores, como a associação de dispositivos IoT a *slices* com requisitos específicos de QoS. Ademais, esse estudo introduziu estratégias biobjetivo, permitindo equilibrar a minimização do número de VANTs com a mitigação de interferências decorrentes da

alocação de dispositivos em configurações semelhantes de SF. As soluções apresentadas evidenciaram ganhos significativos em eficiência espectral, além de demonstrar maior escalabilidade do modelo em cenários realistas.

Com base nesses avanços, este capítulo propõe uma solução fundamentada em MILP que não apenas incorpora os aprendizados consolidados nos estudos anteriores, mas também expande a aplicabilidade das metodologias desenvolvidas. A proposta desta tese busca endereçar lacunas ainda existentes, como a integração mais eficiente de VANTs em cenários de alta demanda e a otimização simultânea de múltiplos objetivos em redes heterogêneas e dinâmicas.

4.2 Modelagem do Sistema

Seja uma rede composta por um conjunto de LoRa-EDs sem mobilidade e distribuídos aleatoriamente, bem como por um conjunto de gateways LoRaWAN instalados em VANTs. Considera-se que cada VANT estabelece comunicação com as BSs 5G, operando em bandas de frequência sub-6 GHz, configuradas com parâmetros compatíveis aos requisitos dos gateways LoRaWAN, sem impor restrições adicionais à operação destes. Nesse contexto, objetiva-se minimizar a quantidade de gateways LoRaWAN necessários para satisfazer os requisitos de QoS estabelecidos, bem como determinar as posições ótimas para a implantação dos VANTs. Para isso, busca-se minimizar o número de posições em um espaço discreto previamente definido para sua instalação.

Seja $\mathcal{K} = \{k_1, \dots, k_k, \dots, k_{|\mathcal{K}|}\}$ o conjunto de LoRa-EDs conectados aos gateways LoRaWAN. Cada LoRa-ED $k \in \mathcal{K}$ está associado a um slice específico pertencente ao conjunto $\mathcal{L} = \{l_1, \dots, l_l, \dots, l_{|\mathcal{L}|}\}$. Cada slice é instanciado na camada física da rede, especificamente nos gateways LoRaWAN, onde a largura de banda é particionada entre os diferentes slices, definidos de acordo com os requisitos de QoS das aplicações IoT. O conjunto de configurações $\mathcal{C} = \{c_1, \dots, c_c, \dots, c_{|\mathcal{C}|}\} \subseteq (\mathcal{F} \times \mathcal{T})$ representa as combinações possíveis entre \mathcal{F} , o conjunto de SFs, e \mathcal{T} , o conjunto de TPs.

O espaço no qual os gateways LoRaWAN podem ser implantados é discretizado no conjunto de posições candidatas $\mathcal{P} = \{p_1 = (x_1, y_1, z_1), p_2 = (x_2, y_2, z_2), \dots, p_{|\mathcal{P}|} = (x_{|\mathcal{P}|}, y_{|\mathcal{P}|}, z_{|\mathcal{P}|})\}$, composto por pontos equidistantes e uniformemente distribuídos ao longo de três eixos perpendiculares, com espaçamento fixo d . Denominam-se posições candidatas os pontos pertencentes a \mathcal{P} que podem ser selecionados pelo otimizador para a implantação dos VANTs.

Para que os requisitos de QoS sejam satisfeitos em cada slice $l \in \mathcal{L}$, a configuração da rede deve considerar parâmetros de taxa de transmissão e atraso, os quais são críticos para garantir o desempenho das aplicações. O QoS é, então, caracterizado por

$$QoS_{c,l} = \overline{r_{c,l}} + (1 - \overline{d_{c,l}}), \quad (4-1)$$

onde $\overline{r_{c,l}}$ e $\overline{d_{c,l}}$ representam, respectivamente, os valores normalizados da taxa de transmissão e do atraso, obtidos pela divisão pelos maiores valores admissíveis de taxa de dados e atraso em enlaces **LoRa**.

A taxa de transmissão alcançada pela configuração $c \in \mathcal{C}$, para o *slice* $l \in \mathcal{L}$, é definida por

$$r_{c,l} = SF(c) \cdot \frac{b_l}{2^{SF(c)}} \cdot CR \quad (\text{bits/s}), \quad (4-2)$$

em que $SF(c)$ representa o fator de espalhamento associado à configuração c , o qual determina a taxa de bits por símbolo, b_l é a largura de banda alocada no *gateway* **LoRaWAN** para o canal do *slice* $l \in \mathcal{L}$, e CR é a taxa de codificação associada ao campo de *Cyclic Redundancy Check* (**CRC**), com valores típicos variando entre $\frac{4}{5}$ e $\frac{4}{8}$.

O atraso de transmissão associado à configuração $c \in \mathcal{C}$ e ao *slice* $l \in \mathcal{L}$ é definido por

$$d_{c,l} = \frac{s}{r_{c,l}}, \quad (4-3)$$

em que s representa o tamanho, em bits, do pacote transmitido por um dispositivo $k \in \mathcal{K}$ por meio de um *gateway* **LoRaWAN**.

No sentido *uplink*, o sinal transmitido por um **LoRa-ED** deve ser recebido por um *gateway* **LoRaWAN** com potência superior à sensibilidade de seu receptor, considerando o *spreading factor* (**SF**) utilizado, caso contrário, o sinal não poderá ser decodificado. Esse sinal está sujeito à degradação ao longo do percurso, devido a fenômenos como reflexão superficial, absorção e refração. Para representar a atenuação em função da distância, adota-se um modelo de perda de trajetória por distância logarítmica (*Log-Distance Path Loss* (**LDPL**)), onde $PL_{(p,k)}$ é a potência perdida pelo sinal transmitido pelo dispositivo $k \in \mathcal{K}$ até o *gateway* **LoRaWAN** posicionado na posição candidata $p \in \mathcal{P}$, que é definida por

$$PL_{(p,k)} = 10 \cdot n \cdot \log_{10} \left(\frac{d_{(p,k)}}{d_0} \right) + P_{r0}, \quad (4-4)$$

em que $d_{(p,k)}$ representa a distância entre a posição $p \in \mathcal{P}$ e o dispositivo $k \in \mathcal{K}$, os parâmetros n , d_0 e P_{r0} correspondem, respectivamente, ao expoente de perda do modelo, à distância de referência e à potência de recepção nessa referência, obtidos empiricamente.

4.3 Formulação do Problema

Nesta seção, apresenta-se a evolução na formulação do problema de otimização, com foco no posicionamento estratégico dos **VANTs**, na alocação eficiente dos dispositivos **LoRa-ED** e na configuração adequada dos parâmetros de comunicação. As formulações propostas visam garantir o cumprimento dos requisitos de **QoS**, ao mesmo tempo em que minimizam o número de **VANTs** implantados e otimizam o desempenho da rede em termos de eficiência e confiabilidade. Inicialmente, o problema foi modelado com uma função objetivo única, voltada à minimização da quantidade de **VANTs**, à definição de suas posições ideais e à associação dos **LoRa-EDs** aos respectivos *slices*. No entanto, verificou-se que essa abordagem mono-objetivo acarretava sobrecarga em determinadas fatias e canais, sobretudo pela concentração de dispositivos em configurações de **SF** aparentemente mais vantajosas, o que resultava em degradação do **QoS** global.

Com a evolução da pesquisa, propôs-se a incorporação de um segundo objetivo associado à variação no número de dispositivos atribuídos a um mesmo **SF**. Essa extensão resultou em uma formulação biobjetivo, permitindo equilibrar a minimização da quantidade de **VANTs** com a mitigação de interferências decorrentes da alocação desigual de dispositivos entre as diferentes configurações de **SF**, o que proporcionou melhorias significativas no desempenho global e na eficiência espectral da rede.

Modelos de Otimização

Para tratar a alocação eficiente de **VANTs** em redes **IoT**, diferentes modelos de otimização foram desenvolvidos ao longo desta pesquisa. Inicialmente, a formulação mono-objetivo centrava-se na minimização do número de **VANTs**, assegurando conectividade e qualidade de serviço para os dispositivos finais. No entanto, as limitações observadas nessa abordagem, como a concentração excessiva de dispositivos utilizando o mesmo **SF**, motivaram a evolução para uma formulação biobjetivo. Esse aprimoramento possibilitou equilibrar a eficiência operacional com a mitigação de interferências, considerando cenários mais desafiadores e aderentes à realidade. A seguir, são descritos os dois modelos, com ênfase em suas características e contribuições específicas.

Abordagem mono-objetivo

A abordagem mono-objetivo proposta por [Silva et al. 2023] visa minimizar a quantidade de **VANTs** implantados para atendimento aos dispositivos **LoRa-ED** distribuídos em um ambiente **IoT** não-**3GPP**, considerando as configurações de comunicação e as

restrições de conectividade. Nesse contexto, a função objetivo é definida como

$$\text{minimizar} \quad \sum_{p \in \mathcal{P}} \left[\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \frac{x_{k,c}^p}{|\mathcal{K}|} \right], \quad (4-5)$$

em que $x_{k,c}^p \in \{0, 1\}$ é uma variável binária que indica a presença de um **VANT** com *gateway* **LoRaWAN** posicionado em $p \in \mathcal{P}$ para atendimento ao dispositivo $k \in \mathcal{K}$, utilizando a configuração $c \in \mathcal{C}$.

O modelo está sujeito às seguintes restrições:

$$\left[\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \frac{x_{k,c}^p}{|\mathcal{K}|} \right] \leq 1, \quad \forall p \in \mathcal{P}, \quad (4-6)$$

a qual assegura que cada posição $p \in \mathcal{P}$ possa ser ocupada por, no máximo, um **VANT**.

Considerando que, em redes **LoRaWAN**, um mesmo dispositivo pode ter seus dados recebidos por múltiplos *gateways*, e que a conexão pode ocorrer com diferentes combinações de **SF** e **TP**, impõe-se a restrição

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x_{k,c}^p = 1, \quad \forall k \in \mathcal{K}. \quad (4-7)$$

Essa restrição assegura que cada dispositivo $k \in \mathcal{K}$ seja associado a exatamente um *gateway* **LoRaWAN** posicionado em $p \in \mathcal{P}$, utilizando uma única configuração $c \in \mathcal{C}$.

A soma do tráfego de *uplink* gerado pelos dispositivos associados a um *slice* $l \in \mathcal{L}$ não deve exceder a capacidade máxima de tráfego suportada por um *gateway* **LoRaWAN** posicionado em $p \in \mathcal{P}$. Assim, impõe-se a restrição

$$\sum_{p \in \mathcal{P}} \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} S_{k,l} \cdot x_{k,c}^p \cdot R_k \leq R_l^{\max}, \quad \forall l \in \mathcal{L}, \quad (4-8)$$

a qual garante que o somatório do tráfego de *uplink* R_k gerado por todos os dispositivos $k \in \mathcal{K}$, associados ao *slice* $l \in \mathcal{L}$, não ultrapasse a capacidade máxima R_l^{\max} alocada para esse *slice* em um *gateway* **LoRaWAN**.

A ligação binária $S(k, l)$ garante que o dispositivo $k \in \mathcal{K}$ esteja associado ao *slice* $l \in \mathcal{L}$. Além disso, impõe-se a restrição

$$x_{k,c}^p \leq I_{k,c}^p, \quad \forall p \in \mathcal{P}, \forall k \in \mathcal{K}, \forall c \in \mathcal{C}, \quad (4-9)$$

na qual $I_{k,c}^p \in \{0, 1\}$ é uma variável binária que indica se o dispositivo $k \in \mathcal{K}$ é capaz de se comunicar com um *gateway* **LoRaWAN** posicionado na coordenada $p \in \mathcal{P}$, utilizando a configuração $c \in \mathcal{C}$.

Para assegurar que os níveis de **QoS** estejam em conformidade com os limites

estabelecidos nos **SLAs**, define-se a restrição

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x_{k,c}^p \cdot \mathcal{S}(k, l) \cdot QoS_k \geq \rho_l^{QoS}, \quad \forall l \in \mathcal{L}, \forall k \in \mathcal{K}, \quad (4-10)$$

na qual QoS_k representa o valor do indicador de qualidade de serviço para o dispositivo $k \in \mathcal{K}$ conectado a um *gateway* **LoRaWAN**; a associação do dispositivo ao *slice* $l \in \mathcal{L}$ é indicada por $\mathcal{S}(k, l)$; e ρ_l^{QoS} é o parâmetro que define o limite inferior de **QoS** exigido para o *slice* l , conforme o respectivo **SLA**.

Abordagem Biobjetivo

A abordagem biobjetivo proposta em [Pires-Jr et al. 2024, Silva et al. 2024] tem como objetivo minimizar simultaneamente a quantidade de **VANTs** implantados e a concentração de dispositivos em uma mesma configuração de **SF**, a fim de reduzir interferências e aumentar a eficiência do sistema.

Seja $x_{k,c}^p \in \{0, 1\}$ uma variável binária de decisão, a qual indica que um *gateway* **LoRaWAN** embarcado em um **VANT** foi posicionado na posição candidata $p \in \mathcal{P}$ para atender ao dispositivo $k \in \mathcal{K}$ utilizando a configuração $c \in \mathcal{C}$. Considera-se que um **VANT** está implantado sempre que ao menos uma posição candidata $p \in \mathcal{P}$ for atribuída a qualquer dispositivo, independentemente da configuração adotada. Assim, define-se

$$N_{\text{VANTs}} = \sum_{p \in \mathcal{P}} \left[\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} \frac{x_{k,c}^p}{|\mathcal{K}|} \right]. \quad (4-11)$$

Em uma rede **LoRaWAN**, os **LoRa-EDs** podem causar interferência mútua quando seus sinais são recebidos simultaneamente, com potência suficiente, por um mesmo *gateway* **LoRaWAN**. Essa interferência torna-se mais significativa quando os dispositivos utilizam o mesmo **SF** [Magrin, Centenaro e Vangelista 2017]. Dessa forma, quanto menor o número de dispositivos transmitindo com uma mesma configuração de **SF**, menor é a probabilidade de sobreposição de sinais no *gateway* **LoRaWAN**.

Consequentemente, define-se o melhor posicionamento dos *gateways* **LoRaWAN** como aquele que resulta na menor concentração de **LoRa-EDs** utilizando um único **SF**. Para esse fim, introduz-se a variável y , definida como

$$y \geq \sum_{c \in \mathcal{C}_{\text{SF}}} \sum_{p \in \mathcal{P}} \sum_{k \in \mathcal{K}} x_{k,c}^p, \quad \forall \text{SF} \in \mathcal{F}. \quad (4-12)$$

Nesse contexto, define-se a função objetivo do modelo biobjetivo como

$$\text{minimizar} \quad \alpha \cdot N_{\text{VANTs}} + \beta \cdot y, \quad (4-13)$$

na qual a importância relativa de cada objetivo pode ser ajustada por meio dos pesos α e β . Especificamente, α corresponde ao peso atribuído à minimização da quantidade de **VANTs** implantados, enquanto β está associado à redução da concentração de dispositivos em uma mesma configuração de **SF**, promovendo maior equilíbrio na distribuição dos dispositivos. O ajuste desses pesos permite a personalização do processo de otimização conforme as exigências do cenário analisado. As restrições do modelo biobjetivo incluem:

Atribuição única por dispositivo - para cada dispositivo $k \in \mathcal{K}$, exatamente uma combinação de configuração $c \in \mathcal{C}$ e posição $p \in \mathcal{P}$ deve ser selecionada, conforme a restrição

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x_{k,c}^p = 1, \quad \forall k \in \mathcal{K}. \quad (4-14)$$

Capacidade do gateway LoRaWAN — o tráfego total de *uplink*, denotado por R_k , gerado pelos dispositivos $k \in \mathcal{K}$, não deve exceder a capacidade máxima de tráfego R_l^{\max} reservada para o *slice* $l \in \mathcal{L}$ em cada *gateway* LoRaWAN posicionado em $p \in \mathcal{P}$. Para isso, impõe-se a seguinte restrição

$$\sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} S(k, l) \cdot x_{k,c}^p \cdot R_k \leq R_l^{\max}, \quad \forall l \in \mathcal{L}, \forall p \in \mathcal{P}, \quad (4-15)$$

em que $S(k, l) \in \{0, 1\}$ é uma função indicadora que retorna 1 quando o dispositivo k está associado ao *slice* l , e 0 caso contrário.

Garantia de QoS — é necessário assegurar que o nível de **QoS** de cada dispositivo associado ao *slice* $l \in \mathcal{L}$ esteja em conformidade com o limite inferior ρ_l^{QoS} estabelecido nos **SLAs**, conforme expresso em

$$\sum_{p \in \mathcal{P}} \sum_{c \in \mathcal{C}} x_{k,c}^p \cdot S(k, l) \cdot QoS_{c,l} \geq \rho_l^{\text{QoS}}, \quad \forall l \in \mathcal{L}, \forall k \in \mathcal{K}. \quad (4-16)$$

Atenuação de sinal — a associação entre um dispositivo **LoRa-ED** e um *gateway* LoRaWAN só é permitida se a potência recebida for suficiente, considerando a perda de propagação. Tal condição é modelada por

$$\sum_{p \in \mathcal{P}} \sum_{k \in \mathcal{K}} \sum_{c \in \mathcal{C}} x_{k,c}^p [P_{tx}(c) - PL_{(p,k)}] \geq S_{rx}(c), \quad (4-17)$$

em que $P_{tx}(c)$ representa a potência de transmissão associada à configuração $c \in \mathcal{C}$, $PL_{(p,k)}$ é a perda de propagação entre o dispositivo k e a posição p , e $S_{rx}(c)$ denota a sensibilidade do receptor do *gateway* LoRaWAN para a configuração c .

Nesta etapa da modelagem, adota-se um modelo determinístico de perda de propagação do tipo **LDPL**, que considera exclusivamente a atenuação do sinal em função da distância. Desta forma, opta-se deliberadamente por suprimir os efeitos de *shadowing*, a

fim de preservar a estrutura determinística da formulação e garantir sua tratabilidade computacional no contexto da MILP. A introdução de componentes estocásticos associados ao *shadowing*, acarretaria aumento significativo na complexidade do modelo, comprometendo a resolubilidade exata e os tempos de execução. Esta escolha está em consonância com a abordagem metodológica híbrida proposta, uma vez que a segunda fase, baseada em algoritmos de DRL, é responsável por incorporar de forma implícita os efeitos dinâmicos e variáveis do ambiente, por meio de simulações mais realistas conduzidas no ns-3.

Assim, define-se o problema de otimização para a implantação de VANTs conforme a função objetivo apresentada na Equação 4-13, sujeita às restrições expressas nas Equações (4-14)–(4-17). A Tabela 4.1 apresenta um resumo da notação utilizada na formalização do modelo MILP.

Tabela 4.1: Notação utilizada na formalização da solução por MILP. [Criada pelo Autor.]

	Notação	Descrição
Conjuntos	\mathcal{P}	Conjunto de posições candidatas discretas
	\mathcal{K}	Conjunto de <i>LoRa End Devices</i>
	\mathcal{L}	Conjunto de <i>slices</i>
	\mathcal{F}	Conjunto de opções de <i>spreading factor</i>
	\mathcal{T}	Conjunto de opções de potência de transmissão
	\mathcal{C}	Conjunto de configurações, tal que $\mathcal{C} \subseteq \mathcal{F} \times \mathcal{T}$
Parâmetros	$\mathcal{S}(k, l)$	Associação binária do dispositivo k ao <i>slice</i> l
	R_k	Tráfego de <i>uplink</i> gerado pelo dispositivo k
	R_l^{\max}	Capacidade máxima de tráfego permitida no <i>slice</i> l
	$QoS_{c,l}$	Valor de QoS para a configuração c do <i>slice</i> l
	ρ_l^{QoS}	Limite inferior de QoS exigido para o <i>slice</i> l
	$r_{c,l}$	Taxa de transmissão (bits/s) para a configuração c no <i>slice</i> l
	$d_{c,l}$	Atraso (s) para a configuração c no <i>slice</i> l
	$SF(c)$	<i>Spreading factor</i> da configuração c
	$P_{tx}(c)$	Potência de transmissão da configuração c
	$S_{rx}(c)$	Sensibilidade do receptor para a configuração c
	$PL_{(p,k)}$	Perda de potência do sinal do dispositivo k até a posição p
	$d(p, k)$	Distância entre o dispositivo k e a posição p do VANT
	d_0	Distância de referência para o cálculo da perda de caminho
	P_{r0}	Potência de recepção de referência a d_0
	b_l	Largura de banda atribuída ao <i>slice</i> l no <i>gateway LoRaWAN</i>
L	Tamanho do pacote transmitido pelos dispositivos, em bits	
Variáveis	$x_{k,c}^p$	Indica se o dispositivo k é atendido em p usando configuração c
	y	Número máximo de dispositivos usando uma mesma configuração de SF
	N_{VANTs}	Quantidade total de VANTs implantados
	α	Peso atribuído à minimização do número de VANTs na função objetivo
	β	Peso atribuído ao balanceamento da distribuição dos dispositivos entre SFs

4.4 Avaliações

Nesta seção, realiza-se a análise de desempenho das soluções desenvolvidas com base na modelagem por **MILP**, aplicada ao problema de alocação dinâmica de recursos em redes **IoT** não-**3GPP** com suporte de **VANTs**. A solução proposta visa minimizar a quantidade de **VANTs** a serem implantados, otimizar seus posicionamentos, definir as configurações dos **LoRa-EDs** e aloca-los adequadamente aos respectivos *slices*. A abordagem de otimização por **MILP** possibilita explorar diferentes cenários e configurações, permitindo avaliar sua eficácia frente aos requisitos de **QoS** estabelecidos.

Os experimentos conduzidos nesta pesquisa foram concebidos para representar um ambiente urbano realista, utilizando parâmetros baseados nas regulamentações da aviação civil e nas características específicas das redes **LoRaWAN** e **5G**. O objetivo é avaliar a aplicabilidade prática do modelo **MILP** proposto em cenários complexos, verificando sua capacidade de atender aos requisitos de **QoS** em redes **IoT** heterogêneas.

As avaliações têm por finalidade verificar a eficiência das soluções obtidas em diferentes topologias e configurações de rede, considerando variáveis como a densidade de dispositivos, o número de **VANTs** requeridos, as configurações adotadas pelos **LoRa-EDs**, entre outros aspectos. São apresentados os resultados das simulações que ilustram o impacto das decisões tomadas pelo modelo sobre métricas como taxa de dados, atraso e **QoS**. Adicionalmente, discute-se as limitações das abordagens propostas e os cenários em que sua aplicação se mostra mais eficaz.

A metodologia do processo de avaliação dos métodos de otimização propostos adotada neste trabalho segue o fluxo ilustrado na Figura 4.1, que apresenta as etapas do processo de desenvolvimento, desde a geração de dados até a configuração final do ambiente de simulação e a obtenção dos resultados. As etapas são detalhadas nas seções subsequentes.

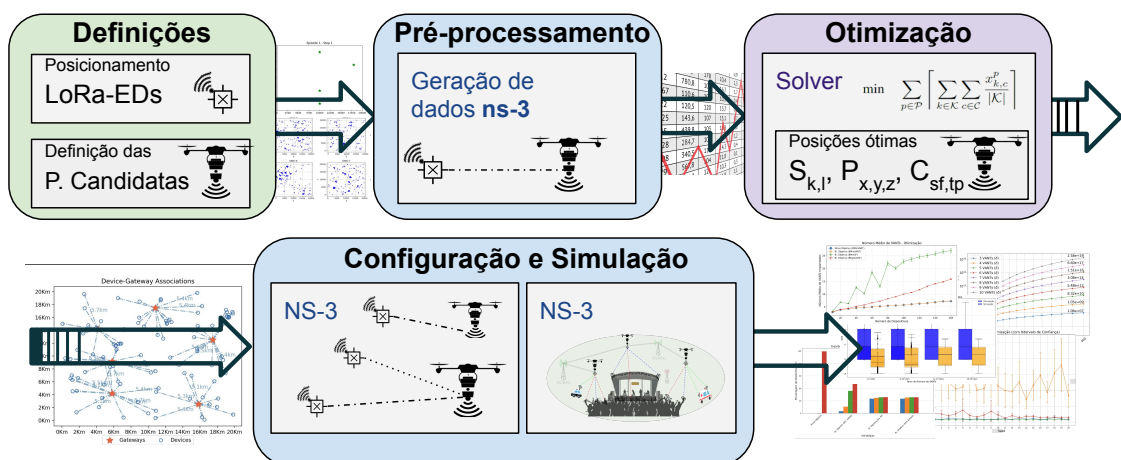


Figura 4.1: Metodologia do processo de avaliação dos métodos de otimização propostos. [Criada pelo Autor.]

Definições

Nesta etapa, são definidas as posições iniciais dos dispositivos e o conjunto de posições candidatas para alimentar a etapa de pré-processamento dos dados no ns-3.

Definição do número de dispositivos

Inicialmente, os dispositivos LoRa, representados pelo conjunto \mathcal{K} , foram distribuídos em uma área urbana externa de 20 km², considerando um cenário *Urban Macro* (UMa), conforme especificado pelo padrão 3GPP. A escolha de uma área com essa dimensão visa explorar os limites operacionais das redes LPWANs, especialmente em ambientes urbanos densamente povoados. O posicionamento dos dispositivos foi realizado com base em uma formulação originalmente desenvolvida para modelagem de tráfego de dados em redes urbanas [Lee et al. 2014], que foi adaptada neste trabalho com o propósito de representar a distribuição espacial dos dispositivos em cenários urbanos complexos.

A distribuição adotada segue um padrão log-normal, que se destaca por sua capacidade de capturar variações topológicas e ajustar a densidade de dispositivos de acordo com padrões dinâmicos de ocupação territorial. Essa abordagem permite uma representação mais realista da presença de dispositivos em ambientes urbanos e rurais. A Figura 4.2 ilustra o posicionamento de 200 dispositivos em uma série de vinte instâncias geradas a partir de distribuições log-normais.

O padrão log-normal é amplamente utilizado em estudos de distribuição espacial devido à sua flexibilidade em modelar fenômenos com alta variabilidade e assimetria, características comuns na distribuição de dispositivos IoT. Essas propriedades são especialmente relevantes em ambientes urbanos, onde a densidade de dispositivos pode variar significativamente em função da topografia, da presença de edificações e da mobilidade dos usuários. Assim, este trabalho adota tal modelo para representar a distribuição dos LoRa-EDs em um cenário urbano de alta densidade.

As Equações 4-18 e 4-19 descrevem o modelo utilizado para capturar padrões espaciais realistas, combinando a distribuição log-normal com uma mistura harmônica de distribuições para refletir a complexidade dos cenários urbanos. A densidade de dispositivos é definida pela função

$$\rho_{wh} = e^{\mu + \sigma \cdot X_{wh}}, \quad (4-18)$$

em que μ representa a média logarítmica da distribuição, σ o desvio padrão logarítmico e X_{wh} uma variável aleatória padronizada, obtida a partir da mistura log-normal definida por

$$X_{wh} = \frac{2}{\sqrt{L}} \sum_{l=1}^L \cos(f_{x,l} \cdot x_w + \phi_l) \cdot \cos(f_{y,l} \cdot y_h + \psi_l), \quad (4-19)$$

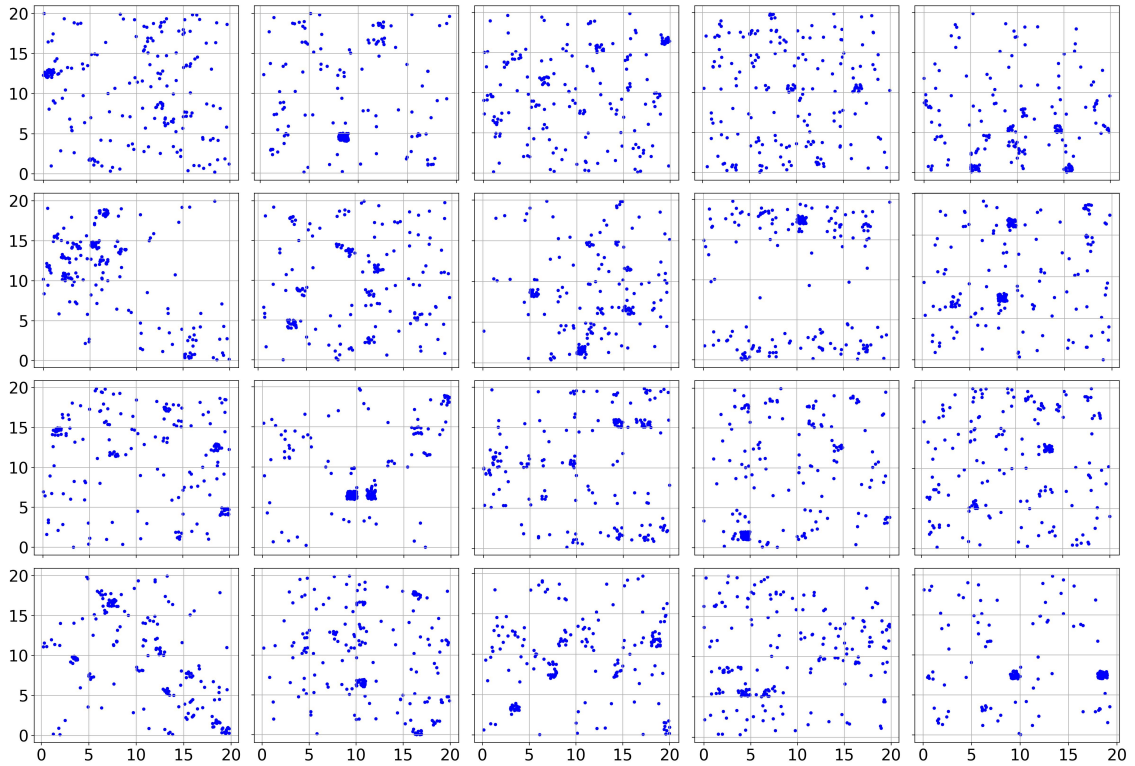


Figura 4.2: Ilustração da distribuição log-normal de dispositivos em uma área de 20 km². [Criada pelo Autor.]

em que L é o número de componentes da mistura, $f_{x,l}$ e $f_{y,l}$ são as frequências angulares nas direções x e y , respectivamente, e ϕ_l, ψ_l representam fases aleatórias.

Essa abordagem possibilita uma modelagem mais realista da ocupação espacial dos dispositivos na rede, ao incorporar variações naturais de densidade no interior da área simulada.

Definição do número de posições candidatas

Na etapa seguinte, definiram-se as posições candidatas para a implantação dos **VANTs**, representadas pelo conjunto \mathcal{P} . O número total de posições foi inicialmente estabelecido como $\mathcal{P} = n^2$, com $n \in \mathbb{N}$ e $1 \leq n \leq 20$, organizadas de forma uniformemente espaçada. Essas posições foram dispostas em uma malha tridimensional regular, com pontos equidistantes fixados a uma altitude de 45 metros, em conformidade com as diretrizes operacionais da **ANAC**. A Figura 4.3 ilustra a distribuição espacial das posições candidatas simuladas.

Foram considerados cenários com diferentes quantidades de posições candidatas, com o objetivo de avaliar o impacto dessa variável na distribuição dos **VANTs**, nos níveis

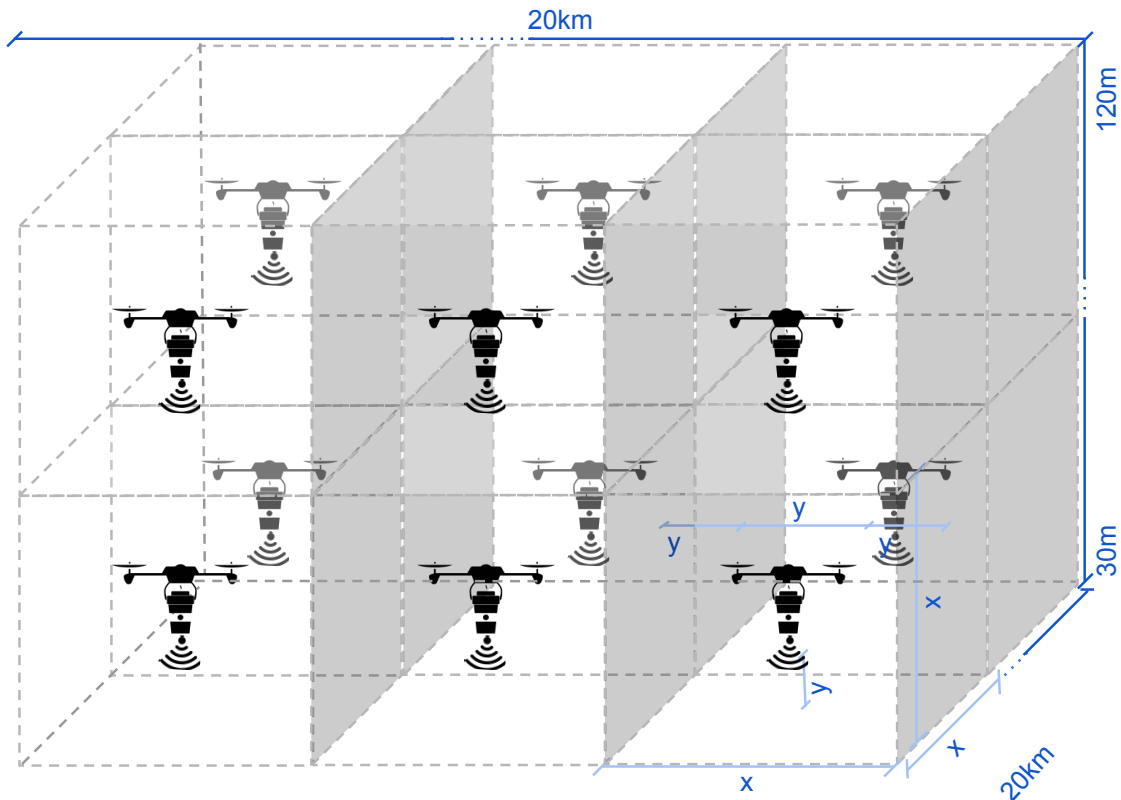


Figura 4.3: Ilustração das posições candidatas para implantação de VANTs. [Criada pelo Autor.]

de QoS alcançados e no consumo computacional associado à resolução do modelo de otimização.

Para delimitação da amplitude dos experimentos, executou-se o solucionador de otimização para todas as combinações de posições candidatas no intervalo de 1×1 a 20×20 . A Figura 4.4 apresenta os resultados dessa análise para a formulação biobjetivo com ênfase na minimização do número de VANTs, a qual também foi replicada para as demais estratégias avaliadas.

Observou-se que, para o caso com apenas uma posição candidata (1×1), todas as execuções do solucionador resultaram em instâncias inviáveis (*infeasible*). Para o caso com 2×2 posições candidatas, apenas os cenários com número reduzido de dispositivos (até 40) apresentaram soluções viáveis. Em contrapartida, para grades de 3×3 e 4×4 , verificou-se um aumento na ocorrência de inviabilidade à medida que crescia a densidade de dispositivos. Por outro lado, em grades com número de posições superior a 16×16 , observou-se aumento significativo no número de execuções interrompidas por extrapolação do tempo limite do solucionador, sem obtenção de solução ótima.

Nos experimentos iniciais, a inviabilidade foi atribuída principalmente à escassez de posições disponíveis para a implantação dos VANTs, ou ainda à limitação na capaci-

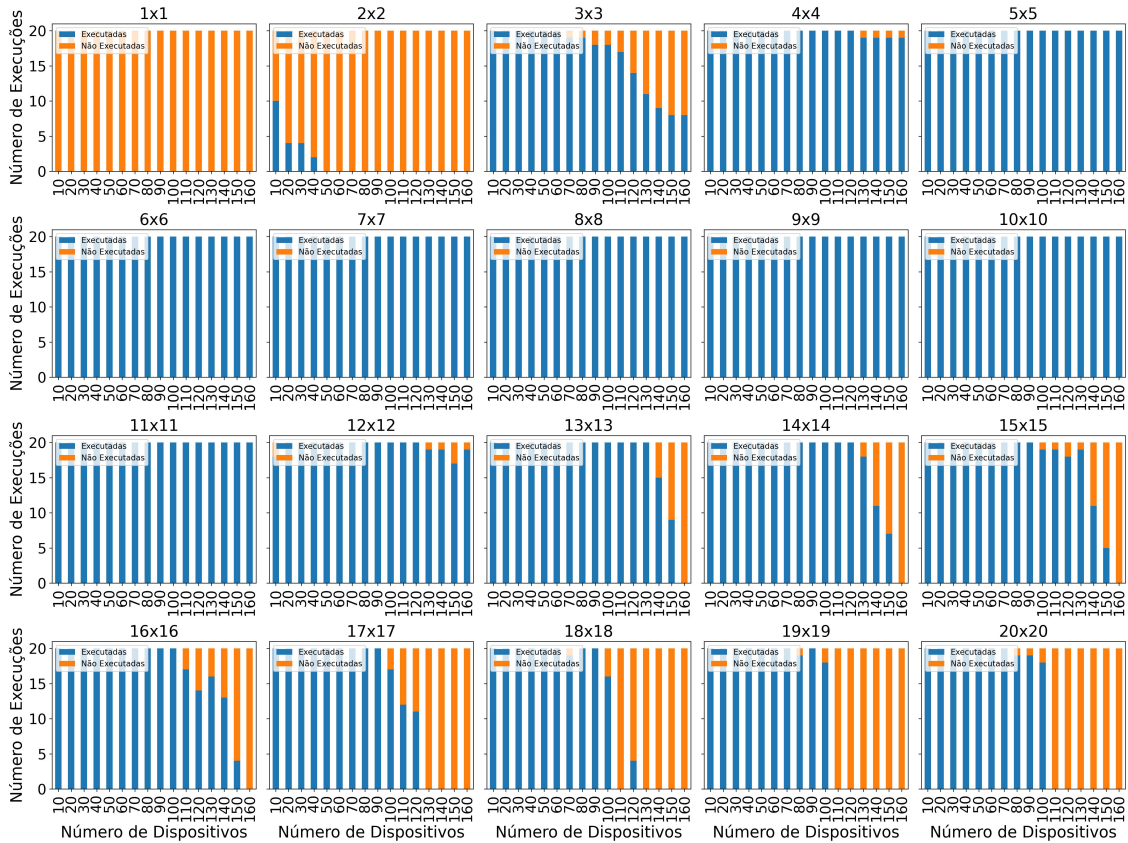


Figura 4.4: *Otimizações realizadas por número de dispositivos e posições candidatas. [Criada pelo Autor.]*

dade de reposicioná-los de forma a atender eficientemente todos os dispositivos sob as restrições impostas. Em contraponto, nos casos com maior densidade de posições (acima de 11×11) e número elevado de dispositivos (acima de 160), a complexidade computacional associada à expansão exponencial do espaço de busca comprometeu a obtenção de soluções ótimas dentro do tempo estipulado.

Com base nessa análise, optou-se por restringir os experimentos aos intervalos de 5×5 até 11×11 posições candidatas e número de dispositivos variando entre 10 e 160.

Para garantir que os requisitos de desempenho fossem atendidos, estabeleceu-se um limite mínimo de **QoS** igual a 0,9 para cada *slice* $l \in \mathcal{L}$. Essa restrição assegura que cada dispositivo seja alocado a um **VANT** de forma otimizada, preservando a qualidade de serviço exigida por aplicações críticas em redes **IoT**. Foram conduzidas vinte simulações independentes, utilizando diferentes sementes e o gerador de números pseudo-aleatórios MRG32k3a. Os resultados são apresentados com intervalo de confiança de 95%.

Pré-processamento

A etapa de pré-processamento tem como objetivo a geração dos dados iniciais a partir de simulações conduzidas no simulador de redes **ns-3**. O **ns-3** recebe como entrada

as localizações dos **LoRa-EDs** e as posições candidatas disponíveis para a implantação dos **VANTs**, realiza a configuração do canal de comunicação, do modelo de propagação e de atraso, define as rodadas de execução e executa o modelo, gerando como saída os dados de comunicação correspondentes a cada par (k, p) , onde $k \in \mathcal{K}$ representa um dispositivo **LoRa-ED** e $p \in \mathcal{P}$ uma posição candidata.

Essa etapa constitui a base para a otimização subsequente, assegurando que as condições simuladas reflitam um ambiente de operação realista. A Tabela 4.2 apresenta os parâmetros utilizados na configuração das simulações no **ns-3**.

Tabela 4.2: Parâmetros utilizados nas simulações no **ns-3**. [Criada pelo Autor.]

Parâmetro	Valor
Modelo de propagação	<i>LogDistancePropagationLossModel</i> , expoente de perdas: 3,36, distância de referência: 10,0 m
Modelo de atraso	<i>ConstantSpeedPropagationDelayModel</i>
Canal de comunicação	<i>LoraChannel</i>
Modelo de mobilidade	<i>ConstantPositionMobilityModel</i>
Região de operação	<i>LorawanMacHelper::EU</i>
Geração e envio de pacotes	Tamanho do pacote: 50 bytes; periodicidade: 30 s
Tempo de simulação	1200 s

Otimização

Na etapa de otimização, o solucionador é executado para cada conjunto de dados gerado na etapa de pré-processamento, retornando como resultado as soluções ótimas. Essas soluções especificam as posições ideais para a implantação dos **VANTs**, as configurações ótimas para os **LoRa-EDs**, bem como os *slices* mais adequados para a associação entre os **LoRa-EDs** e os *gateways* **LoRaWAN** embarcados nos **VANTs**.

Esta seção apresenta, inicialmente, o desenvolvimento do modelo de otimização baseado em **MILP**, seguido do processo de seleção do solucionador e, por fim, da definição das estratégias de simulação adotadas para a condução dos experimentos.

Desenvolvimento do modelo MILP

Na sequência, implementou-se o modelo de otimização descrito na Seção 4.3, utilizando a biblioteca *Pyomo* [Bynum et al. 2021] com a linguagem *Python* para formulação e execução do processo de busca por soluções ótimas. Como solucionador, adotou-se o *Gurobi*, versão 12.0, licenciado para uso acadêmico [Gurobi 2025].

A escolha do *Gurobi* justifica-se principalmente por sua capacidade de paralelização nativa e maior eficiência computacional em relação a alternativas *open source*, como o solucionador *SCIP*, versão 9.1.0 [Bestuzheva et al. 2021]. A Figura 4.5 apresenta uma comparação do tempo de execução entre os dois solucionadores, considerando diferentes tamanhos de instâncias.

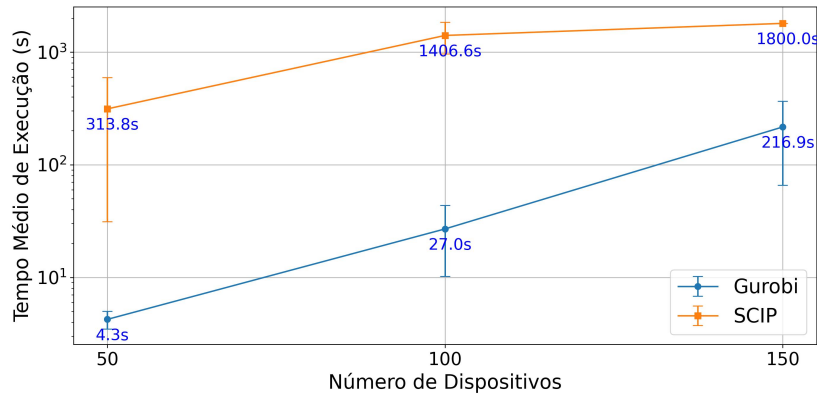


Figura 4.5: Comparação do tempo de execução das otimizações para os solucionadores *Gurobi* e *SCIP*. [Criada pelo Autor.]

Conforme ilustrado na Figura 4.5, o tempo de execução das otimizações utilizando o *Gurobi* foi entre 87,75% e 98,63% inferior ao tempo requerido pelo *SCIP*, considerando os cenários experimentais. As comparações foram realizadas para instâncias com {50, 100, 150} dispositivos *LoRa-ED*, 100 posições candidatas, 10 repetições por configuração, com sementes pseudo-aleatórias geradas pelo algoritmo MRG32k3a, e intervalo de confiança de 95%.

Os experimentos foram executados em uma máquina com processador *AMD Ryzen™ 9 5900X* (24 threads), 64 GB de memória RAM, GPU *NVIDIA GeForce RTX™ 3070 Ti* e sistema operacional Linux Ubuntu 24.04.1 LTS.

Definição das estratégias de experimentação

Quatro estratégias distintas foram definidas para a otimização com base em configurações nos pesos dos parâmetros α e β . A primeira estratégia, **BMinVANT**, é uma otimização biobjetivo com ênfase em minimizar o número de *VANTs* e é configurada com $\alpha > \beta$, para garantir prioridade ao objetivo primário. A segunda estratégia de otimização, **BMinSF**, é biobjetivo com ênfase em minimizar o número máximo de *LoRa-EDs* associados ao mesmo *SF* e é configurada com $\beta > \alpha$, priorizando o objetivo secundário. A terceira estratégia, **BEqVANTSF**, configurada com $\alpha = \beta$, é uma estratégia biobjetivo com ênfase no equilíbrio entre minimizar o número de *VANTs* e a melhor distribuição de dispositivos aos *SFs*. E, finalmente, a estratégia **MMinVANT**, uma otimização mono-

objetivo com minimização do número de **VANTs**, configurada com $\beta = 0$. Essas configurações permitem uma análise comparativa entre as estratégias em diferentes cenários e condições de rede. A Tabela 4.3 sumariza os parâmetros utilizados na otimização.

Tabela 4.3: *Parâmetros utilizados na otimização. [Criada pelo Autor.]*

Parâmetro	Valor
Área	$20Km^2$
\mathcal{P}	Variando entre 4x4 e 16x16
\mathcal{K}	{10, 20, ..., 160}
ρ_l^{QoS}	{0.9}
\mathcal{R}_l^{max}	6835.94 (SF7)
\mathcal{R}_l^{max}	183.11 (SF12)
b_l	125kHz
BMinVANT	$\alpha = 100$ e $\beta = 1$
BMinSF	$\alpha = 1$ e $\beta = 100$
EQVantSF	$\alpha = 50$ e $\beta = 50$
MMinVANT	$\alpha = 100$ e $\beta = 0$
Tamanho do pacote de dados	400 bits
Tempo de simulação	1200 s
Intervalo de confiança	95%
Rodadas de execução	20
Sub-Banda (EU)	863 – 870 MHz
Modelo de propagação	Log-Distance

Execução da otimização

Após a geração dos dados iniciais no ns-3 e a definição dos conjuntos e parâmetros experimentais, a etapa de otimização foi executada. Nesse processo, os valores dos parâmetros $\mathcal{S}(k,l)$, $P_{(x,y,z)}$ e $\mathcal{C}(sf,tp)$ foram determinados por meio do solucionador matemático, com o objetivo de identificar as melhores configurações para, o posicionamento ótimo dos gateways LoRaWAN embarcados nos VANTs, a alocação eficiente dos parâmetros de SF e TP, e a definição dos slices ideais $\mathcal{S}(k,l)$ para associação dos LoRa-EDs.

Configuração e simulação

Após a obtenção das posições e configurações ótimas, o ambiente de simulação foi atualizado com os novos parâmetros resultantes da otimização. Nessa etapa, o simulador ns-3 foi reutilizado para, configurar os dispositivos LoRa-EDs com os valores otimizados, simular o desempenho da rede considerando métricas como latência, atraso médio e taxa de entrega de pacotes, e analisar o impacto das decisões de posicionamento

e alocação sobre o desempenho global da comunicação. Os resultados dessa etapa de simulação são apresentados na Seção 4.5.

A integração entre otimização e simulação permitiu avaliar a solução exata proposta sob diferentes condições de operação, destacando sua robustez e eficácia frente às demandas dinâmicas e heterogêneas das redes IoT. Os resultados obtidos confirmam a aplicabilidade prática do modelo baseado em MILP, bem como sua eficiência na alocação de recursos em ambientes realistas.

4.5 Resultados

Nesta seção, apresentam-se os resultados obtidos a partir das etapas de otimização e simulação conduzidas no simulador ns-3, conforme descrito nas Seções 4.2 e 4.3, bem como nas estratégias de experimentação definidas na Seção 4.4.

Os resultados foram organizados de modo a evidenciar o impacto das diferentes estratégias de otimização, e.g., **MMinVANT**, **BMinVANT**, **BMinSF** e **BEqVANTSF**, sob variações nos parâmetros experimentais, como o número de posições candidatas, a quantidade de dispositivos e seus padrões de distribuição espacial.

As análises realizadas têm como objetivo avaliar a eficácia dos modelos propostos em termos de desempenho global e aderência aos requisitos de QoS, considerando métricas como taxa de dados, atraso e utilização eficiente dos recursos da rede.

Impacto do Número de Posições Candidatas

Esta subseção apresenta os resultados relacionados ao comportamento dos modelos de otimização frente ao aumento no número de posições candidatas disponíveis para a implantação dos VANTs. Partiu-se da hipótese de que uma maior resolução espacial, i.e., um número mais elevado de posições candidatas, proporcionaria maior liberdade ao solucionador para o posicionamento dos VANTs, o que, por sua vez, poderia resultar em melhorias qualitativas nas soluções obtidas.

Com base nessa hipótese, foram conduzidos os experimentos descritos nas seções subsequentes, visando validar o impacto dessa variável nos indicadores de desempenho e QoS.

Impacto do Número de Posições Candidatas no Tempo de Execução das Otimizações

A Figura 4.6 apresenta a relação entre o número de posições candidatas disponíveis e o tempo de execução do modelo de otimização, destacando o impacto do aumento do espaço de busca sobre o desempenho computacional.

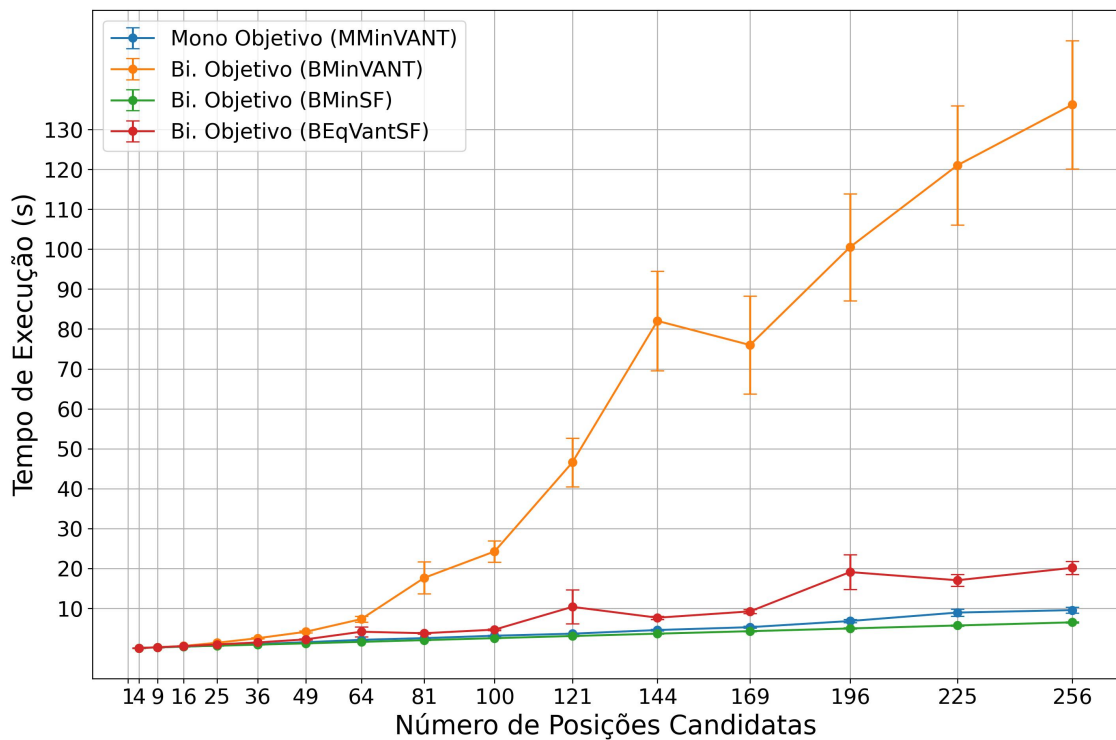


Figura 4.6: Tempo de execução em função do número de posições candidatas para as estratégias experimentadas. [Criada pelo Autor.]

O gráfico demonstra que, para um número reduzido de posições candidatas, até aproximadamente 8×8 , o tempo de execução permanece baixo, variando entre 0 e 10 segundos para todas as estratégias avaliadas. Esse comportamento decorre da baixa complexidade computacional, uma vez que o espaço de busca reduzido permite ao solucionador encontrar soluções ótimas com menor esforço.

No entanto, à medida que o número de posições candidatas cresce, observa-se um aumento no tempo médio de execução, com variações significativas entre as estratégias. A estratégia **BMinVANT**, em particular, apresenta um crescimento exponencial, com tempos médios superiores a 20 segundos para 10×10 posições e ultrapassando 130 segundos para 16×16 . Esse resultado evidencia que a ênfase na minimização do número de **VANTs** impõe um custo computacional elevado, tornando a execução consideravelmente mais complexa à medida que o espaço de busca se amplia.

Em contraste, as estratégias **MMinVANT**, **BMinSF** e **BEqVANTSF** apresentam tempos de execução relativamente estáveis, mesmo diante do aumento do número de posições candidatas. A estratégia **BMinSF**, em especial, destaca-se por apresentar os menores tempos de execução em todo o intervalo analisado, sugerindo que a priorização da distribuição dos dispositivos entre os **SFs** contribui para a redução da complexidade computacional do modelo.

Esse comportamento indica que, embora a minimização do número de **VANTs**

produza ganhos em termos de custo de infraestrutura, ela impõe penalidades computacionais significativas. Já as estratégias que priorizam o equilíbrio de alocação operam com maior eficiência computacional, viabilizando a escalabilidade sem comprometer severamente o tempo de processamento, ainda que à custa de um maior número de **VANTs** implantados.

Outro aspecto relevante diz respeito à variabilidade dos tempos de execução. As barras de erro indicam que a maior dispersão ocorre na estratégia **BMinVANT**, especialmente a partir de 9×9 posições, evidenciando que o tempo de processamento se torna menos previsível conforme cresce a complexidade do problema. Esse comportamento sugere que a solução baseada na minimização do número de **VANTs** é mais sensível às condições iniciais e aos critérios de convergência do solucionador, comprometendo a estabilidade da execução. As demais estratégias, por sua vez, apresentam variações mais contidas, com maior previsibilidade no tempo de processamento.

Diferentemente do que poderia ser esperado, o gráfico não indica estabilização no tempo de execução, sobretudo para a estratégia **BMinVANT**, cujo tempo continua crescendo até o limite superior do intervalo analisado. Esse padrão reforça a natureza combinatória da formulação e evidencia que o modelo não atinge um ponto de saturação computacional nas configurações exploradas. Já as demais estratégias demonstram crescimento mais controlado e tendencialmente linear, o que reforça sua adequação a cenários de maior escala.

Os resultados indicam que a escolha da estratégia de otimização impacta diretamente a viabilidade computacional do modelo. Em cenários nos quais a minimização do número de **VANTs** é essencial, deve-se considerar o alto custo computacional associado e, possivelmente, restringir o número de posições candidatas para manter tempos de execução viáveis. Para aplicações que requerem um compromisso entre custo de implantação e equilíbrio de carga entre os **SFs**, as estratégias **BMinSF** e **BEqVANTSF** revelam-se mais adequadas, por apresentarem maior previsibilidade e controle sobre o tempo de execução.

Essa análise reforça a importância de calibrar cuidadosamente o número de posições candidatas em redes **IoT**, a fim de equilibrar os benefícios da flexibilidade no posicionamento dos **VANTs** com os custos computacionais decorrentes do aumento do espaço de busca. Embora um número maior de posições possa viabilizar soluções mais refinadas, ele também impõe penalidades de desempenho que devem ser consideradas para garantir a viabilidade prática do modelo.

Impacto do Número de Posições Candidatas na QoS

Esta subseção investiga o impacto do número de posições candidatas sobre os níveis de **QoS** alcançados, considerando as diferentes estratégias de otimização adotadas. A análise contempla tanto os resultados teóricos obtidos a partir da otimização quanto

os resultados empíricos provenientes das simulações no ns-3, permitindo comparar o comportamento idealizado com o desempenho prático da rede sob restrições reais de propagação e interferência.

Para garantir a viabilidade das soluções, foi fixado um limiar mínimo de QoS, representado por $\rho_l^{QoS} = 0,9$, assegurando que todas as configurações geradas pelo modelo otimizem a alocação dos dispositivos sem violar os requisitos mínimos de qualidade de serviço. As simulações replicaram fielmente os cenários otimizados, incluindo a distribuição espacial dos dispositivos, o conjunto de posições candidatas e o posicionamento dos VANTs.

A Figura 4.7 apresenta a média de QoS obtida em função do número de posições candidatas, tanto para os resultados da otimização quanto para os resultados das simulações no ns-3. A análise evidencia como a granularidade espacial influencia a alocação dos LoRa-EDs e, conseqüentemente, o desempenho da rede.

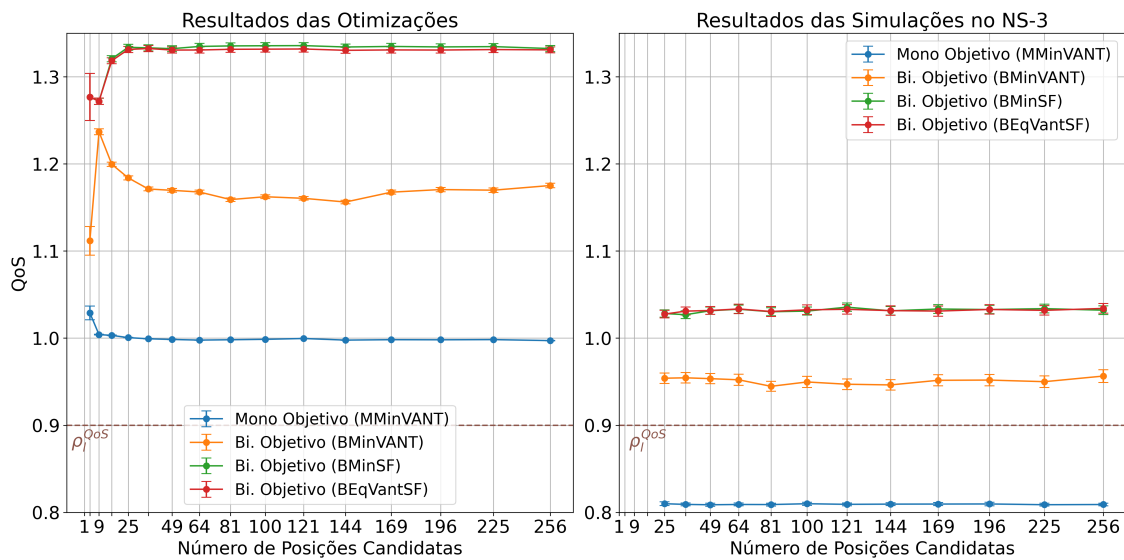


Figura 4.7: Média de QoS em função do número de posições candidatas, considerando otimizações e simulações no ns-3. O limiar mínimo foi fixado em 0,9. [Criada pelo Autor.]

Um dos principais desafios no dimensionamento de redes IoT com suporte de VANTs é minimizar o número de unidades implantadas, visando à redução de custos operacionais (OPEX) e de infraestrutura (CAPEX). A estratégia BMinVANT foi projetada com esse propósito, priorizando a minimização do número de VANTs enquanto mantém a QoS dentro dos limites estabelecidos. No entanto, os resultados revelam que essa abordagem apresenta médias de QoS inferiores às demais estratégias, consequência da redução na flexibilidade de alocação entre diferentes SFs, o que intensifica interferências e reduz a eficiência global da rede.

As estratégias **BMinSF** e **BEqVANTSf**, por sua vez, demonstram melhor desempenho em termos de **QoS**, atingindo valores superiores a 1,30 nas otimizações e acima de 1,00 nas simulações. Esses resultados decorrem da alocação mais equilibrada dos dispositivos entre as configurações de **SF**, o que reduz colisões e melhora o aproveitamento dos recursos espectrais. No entanto, esse desempenho aprimorado vem acompanhado de um aumento no número de **VANTs** implantados, o que pode representar uma limitação prática em termos financeiros e logísticos.

A estratégia **MMinVANT** apresentou os menores níveis de **QoS**, com médias próximas a 1,00 ao longo de todas as configurações. Embora essa abordagem seja vantajosa do ponto de vista de **CAPEX/OPEX**, os resultados indicam que a ausência de um critério de balanceamento de carga prejudica a qualidade de serviço. Isso reforça a relevância de abordagens intermediárias, como a **BMinVANT**, que conciliam custo e desempenho, mantendo a **QoS** dentro de níveis aceitáveis com um número de **VANTs** ainda controlado.

Os resultados das simulações revelam, como esperado, valores de **QoS** inferiores aos obtidos nas otimizações. Essa diferença é explicada pela presença de fatores não modelados diretamente na formulação matemática, como perdas por propagação, interferências e variações no canal. Ainda assim, as tendências gerais observadas se mantêm: as estratégias **BMinSF** e **BEqVANTSf** continuam a apresentar os melhores desempenhos, enquanto **BMinVANT** e **MMinVANT** apresentam médias inferiores.

Destaca-se, no entanto, que a estratégia **BMinVANT** mostrou comportamento mais estável nas simulações, sugerindo que a redução do número de **VANTs**, embora mais sensível ao espaço de busca, pode ser conduzida de forma robusta em cenários reais, desde que configurada adequadamente.

A análise da variabilidade reforça esse entendimento. Nos resultados de otimização, a dispersão dos valores de **QoS** é maior para espaços de busca mais restritos (menor número de posições candidatas), indicando a dificuldade do modelo em encontrar boas soluções sob limitações espaciais. Com o aumento da granularidade, essa variabilidade se reduz. Nas simulações, por outro lado, a dispersão permanece relativamente elevada, independentemente do número de posições candidatas, evidenciando o impacto contínuo de fatores externos ao modelo matemático.

Em termos práticos, observa-se que o aumento no número de posições candidatas impacta positivamente a **QoS** nos cenários com baixa granularidade, mas apresenta ganhos marginais a partir de aproximadamente 100 posições. Isso sugere a existência de um ponto de saturação, além do qual o aumento da resolução espacial não contribui significativamente para a melhoria da qualidade de serviço, mas impõe penalidades computacionais elevadas.

Dessa forma, a minimização do número de **VANTs** deve ser conduzida com cau-

tela, considerando os requisitos de **QoS** e os custos computacionais associados à resolução espacial. Estratégias como **BMinVANT** representam compromissos viáveis entre eficiência operacional e desempenho de rede, enquanto abordagens que priorizam exclusivamente a **QoS** podem inviabilizar implementações reais. A escolha da estratégia de otimização, portanto, deve ser pautada por critérios que conciliem restrições operacionais, limitações financeiras e requisitos de qualidade de serviço, visando garantir a eficiência e a sustentabilidade da rede.

Impacto da Densidade de Dispositivos na QoS

Esta subseção analisa como a densidade de dispositivos afeta a **QoS** nas diferentes estratégias de otimização. O objetivo é compreender os efeitos do aumento no número de **LoRa-EDs** sobre a distribuição do tráfego, a eficiência na alocação dos **VANTs** e, conseqüentemente, sobre a qualidade da comunicação, especialmente considerando a necessidade de reduzir os custos operacionais (**OPEX**) e de infraestrutura (**CAPEX**).

A Figura 4.8 apresenta a relação entre o número de dispositivos na rede e a média de **QoS**, tanto para os resultados obtidos via otimização quanto para aqueles provenientes das simulações realizadas no **ns-3**. Os gráficos evidenciam padrões distintos de degradação da **QoS** conforme a densidade da rede aumenta, permitindo observar as diferenças entre as abordagens mono e biobjetivo.

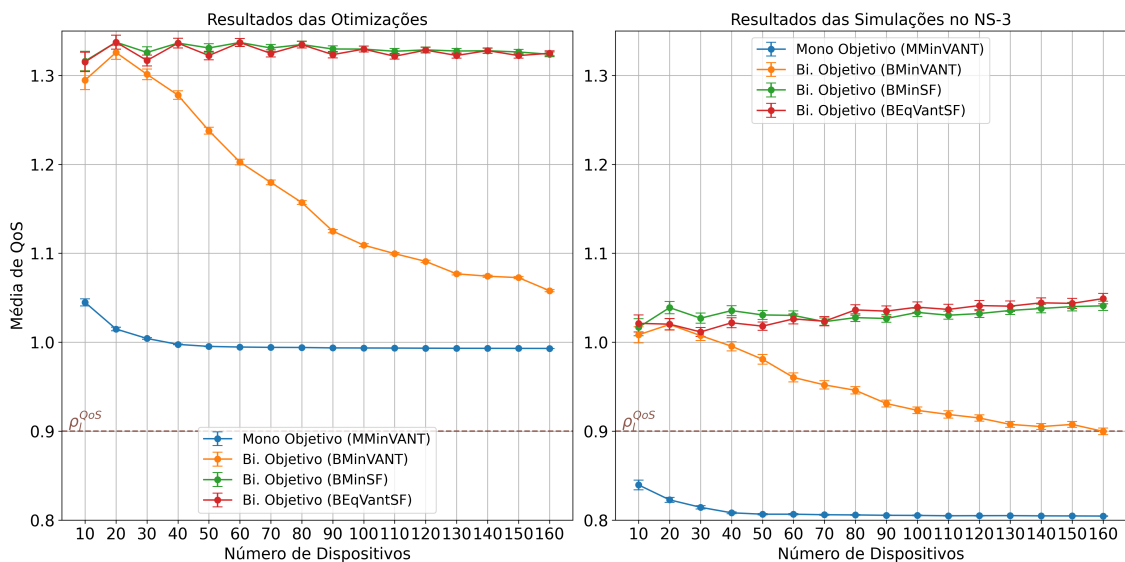


Figura 4.8: Média de QoS em função do número de dispositivos para as estratégias experimentadas. [Criada pelo Autor.]

Nos resultados da otimização, observa-se que as estratégias **BMinSF** e **BEqVANTSF** mantêm valores médios de **QoS** elevados e estáveis, próximos de 1,30, mesmo

com o aumento da densidade de dispositivos. Esse comportamento indica que tais abordagens são mais eficazes na distribuição da carga entre diferentes configurações de **SF**, otimizando o uso dos recursos espectrais e reduzindo interferências.

Em contraste, a estratégia **BMinVANT** apresenta uma degradação progressiva nos níveis de **QoS** à medida que cresce o número de dispositivos. Isso sugere que a minimização do número de **VANTs**, embora eficiente em termos de custo, pode levar à sobrecarga de determinados pontos da rede, comprometendo o desempenho global da comunicação.

A estratégia mono-objetivo **MMinVANT** apresenta os menores valores médios de **QoS** em toda a análise, com médias abaixo de 1,05 na otimização e decaindo para valores próximos de 0,80 nas simulações. Esses resultados reforçam que a ausência de um critério de balanceamento de carga compromete a alocação eficiente dos dispositivos, tornando essa abordagem inadequada para cenários de alta densidade.

Os resultados das simulações no **ns-3**, apresentados no gráfico à direita, revelam uma redução geral dos níveis de **QoS** em relação aos valores otimizados, refletindo os efeitos de condições reais do ambiente de comunicação, como perdas por propagação, interferências e limitações nos enlaces sem fio. Ainda assim, a tendência observada nas otimizações é preservada: **BMinSF** e **BEqVANTSF** mantêm desempenho superior e estável, enquanto **BMinVANT** apresenta deterioração mais acentuada da **QoS** com o aumento da densidade.

Essa análise evidencia a necessidade de balancear a redução do número de **VANTs** com a garantia de qualidade de serviço. Embora **BMinVANT** ofereça um compromisso eficiente em termos de custo, sua adoção deve considerar o impacto sobre a **QoS**, especialmente em aplicações críticas. Para cenários que exigem desempenho elevado e estabilidade, e.g., como redes industriais, saúde conectada ou cidades inteligentes, estratégias como **BMinSF** e **BEqVANTSF** mostram-se mais adequadas, garantindo consistência mesmo em ambientes densamente povoados.

Distribuição de Dispositivos por SF

Esta subseção analisa como os dispositivos **LoRa-ED** são distribuídos entre os diferentes fatores de espalhamento (**SF**) para cada uma das estratégias de otimização avaliadas. A distribuição entre os **SFs** afeta diretamente o desempenho da rede, influenciando os níveis de interferência, a eficiência espectral e a alocação dos recursos disponíveis nos **VANTs**.

A Figura 4.9 apresenta a distribuição percentual de dispositivos em cada valor de **SF**, comparando os resultados das abordagens mono e biobjetivo. As alocações obtidas

refletem as prioridades de cada estratégia e evidenciam seus impactos na eficiência da rede.

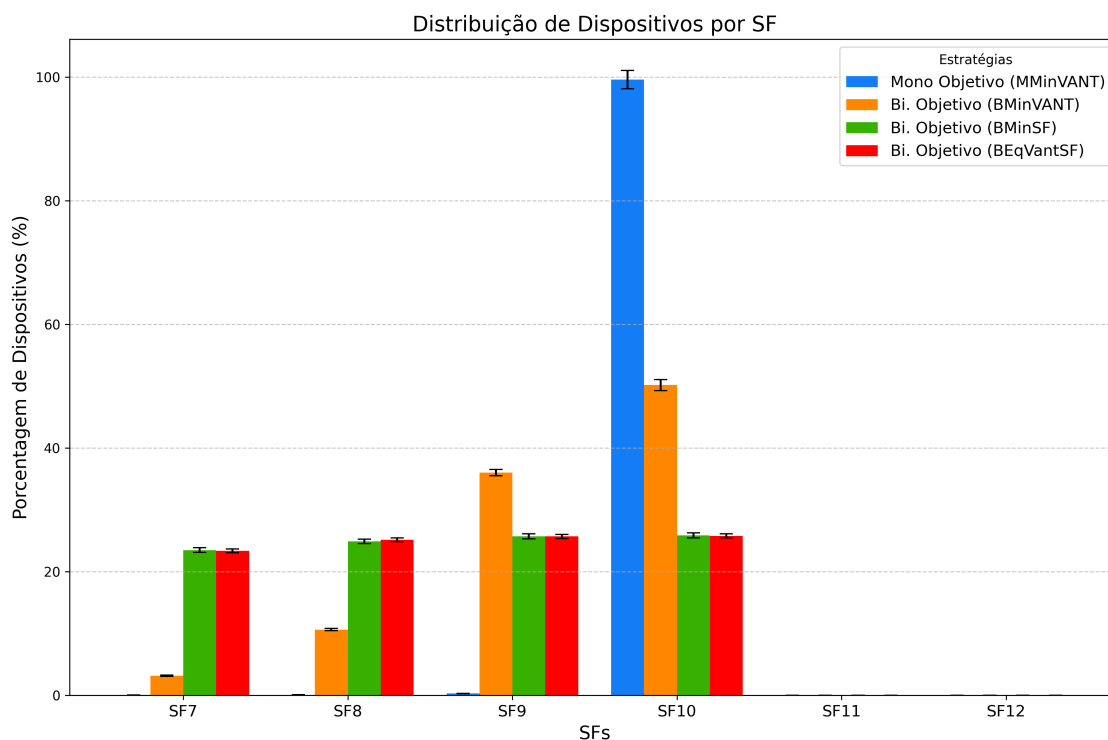


Figura 4.9: Distribuição percentual de dispositivos por SF para cada estratégia de otimização. [Criada pelo Autor.]

Os resultados mostram que a estratégia **MMinVANT**, voltada exclusivamente para a minimização do número de **VANTs**, resulta em uma alocação altamente concentrada no **SF 10**, com quase 100% dos dispositivos utilizando essa configuração. Tal concentração demonstra um desequilíbrio acentuado, que pode elevar os níveis de interferência e comprometer o desempenho da rede. Essa abordagem não considera critérios de balanceamento entre os **SFs**, priorizando unicamente a redução da infraestrutura.

Por outro lado, as estratégias biobjetivo apresentam distribuições mais equilibradas. A estratégia **BMinVANT**, embora ainda concentre parte significativa dos dispositivos nos **SFs 9 e 10**, atenua os efeitos de alocação excessiva em uma única configuração, quando comparada à abordagem mono-objetivo. Esse comportamento revela uma tentativa de mitigar os impactos negativos da concentração, ainda que o foco continue sendo a minimização da quantidade de **VANTs**.

As estratégias **BMinSF** e **BEqVantSF**, que incorporam a distribuição dos dispositivos como componente da função objetivo, resultam em uma alocação mais homogênea entre os **SFs**. Nessas abordagens, observa-se uma distribuição mais equilibrada entre os **SFs 7, 8, 9 e 10**, reduzindo o risco de sobrecarga e interferência. Esse comportamento

favorece a eficiência espectral da rede e diminui a probabilidade de colisões na comunicação, contribuindo para uma operação mais estável e escalável.

A ausência de dispositivos alocados nos SFs 11 e 12 em todas as estratégias analisadas está associada à natureza do problema de otimização. Devido aos seus longos tempos de transmissão e menor eficiência espectral, tais configurações são evitadas automaticamente pelo modelo, que privilegia fatores de espalhamento com maior rendimento.

A análise confirma que estratégias que equilibram a distribuição de dispositivos entre os SFs promovem maior robustez e desempenho da rede. Embora a redução do número de VANTs seja relevante do ponto de vista econômico, o comprometimento da eficiência espectral por meio de alocações desequilibradas pode prejudicar a escalabilidade e a confiabilidade da comunicação. Assim, estratégias como **BMinSF** e **BEqVantSF** demonstram maior adequação para redes IoT em cenários de alta densidade, oferecendo um uso mais eficiente dos recursos disponíveis sem comprometer o desempenho do sistema.

Desempenho das Estratégias nas Simulações

Nesta subseção, avalia-se o desempenho das diferentes estratégias de otimização a partir de três métricas fundamentais para a operação de redes IoT: taxa de entrega de pacotes, vazão média e atraso médio. As Figuras 4.10 e 4.11 ilustram a variação dessas métricas em função do número de dispositivos conectados à rede e da quantidade de posições candidatas, respectivamente.

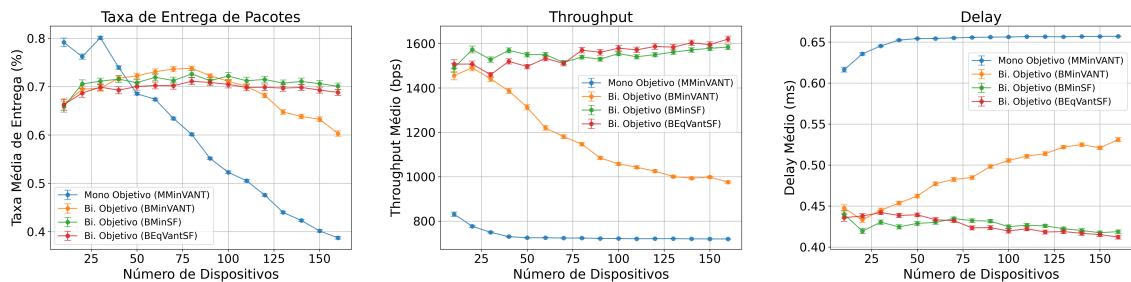


Figura 4.10: Desempenho das estratégias em função do número de dispositivos. [Criada pelo Autor.]

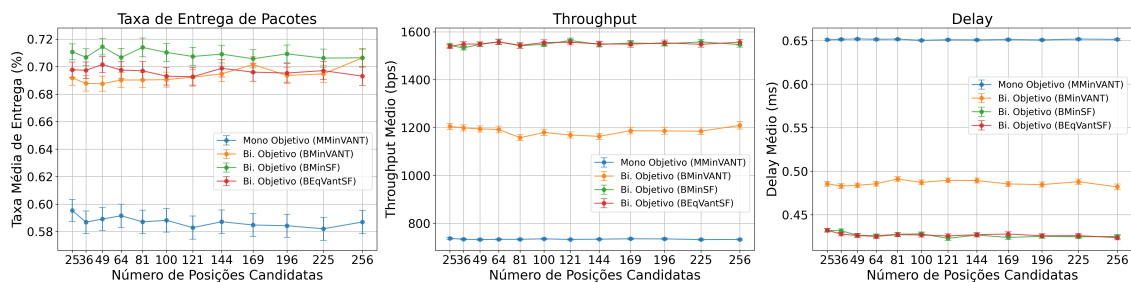


Figura 4.11: Desempenho das estratégias em função do número de posições candidatas. [Criada pelo Autor.]

A análise dos resultados revela que a estratégia **BMinVANT**, focada na minimização do número de **VANTs**, apresenta desempenho inferior nas métricas de taxa de entrega e vazão média à medida que aumenta a densidade de dispositivos. Ainda assim, essa abordagem deve ser considerada sob a perspectiva da redução de custos de implantação e operação, dado seu impacto direto no **CAPEX** e no **OPEX** da rede.

A taxa média de entrega de pacotes apresenta comportamento decrescente para a estratégia **BMinVANT**, atingindo valores inferiores a 50% nos cenários com maior densidade. Em contraste, as estratégias **BMinSF** e **BEqVANTSf** mantêm taxas superiores a 70% em todos os casos, evidenciando que a distribuição equilibrada de dispositivos entre os diferentes fatores de espalhamento (**SFs**) contribui para a redução das colisões e o aumento da eficiência da rede.

Com relação à vazão média, observa-se padrão semelhante. As estratégias **BMinSF** e **BEqVANTSf** alcançam valores superiores, estabilizando-se acima de 1500 bps, enquanto **BMinVANT** sofre degradação progressiva com o aumento do número de dispositivos. Essa redução de desempenho está associada à maior competição pelo espectro disponível, intensificada pela menor quantidade de **VANTs** ativos para suportar a demanda crescente.

O atraso médio também se configura como fator crítico para a estratégia **BMinVANT**. Em cenários com alta densidade, o atraso médio ultrapassa 0,6 ms, ao passo que as demais estratégias mantêm valores consideravelmente mais baixos e estáveis. Este resultado reforça a eficácia das abordagens que promovem balanceamento entre os **SFs** e os pontos de acesso.

Apesar das limitações observadas nas métricas de desempenho, a estratégia **BMinVANT** permanece relevante quando a principal meta é a redução da infraestrutura. Em aplicações nas quais a minimização do número de **VANTs** é mandatória, essa abordagem pode ser viável, desde que haja flexibilidade nos requisitos de **QoS**. Alternativamente, um compromisso mais equilibrado entre custo e desempenho pode ser alcançado com as estratégias **BMinSF** e **BEqVANTSf**, que promovem distribuição de carga mais eficiente, ainda que com um número maior de **VANTs**.

Esses resultados reforçam a importância do balanceamento entre múltiplos critérios de otimização. A escolha da estratégia mais adequada deve considerar simultaneamente os indicadores de desempenho da rede e os custos de implantação, de forma a garantir a viabilidade técnica e econômica da solução em cenários reais.

Análise da Distribuição da QoS em Função do Número de VANTs Implantados

Esta subsecção analisa a relação entre os níveis de QoS e a quantidade de VANTs implantados, com base nos resultados obtidos por meio das estratégias de otimização e nas simulações realizadas no ns-3. A Figura 4.12 apresenta a distribuição da QoS em faixas de número de VANTs, permitindo uma análise comparativa entre os cenários otimizados e os resultados práticos obtidos na simulação.

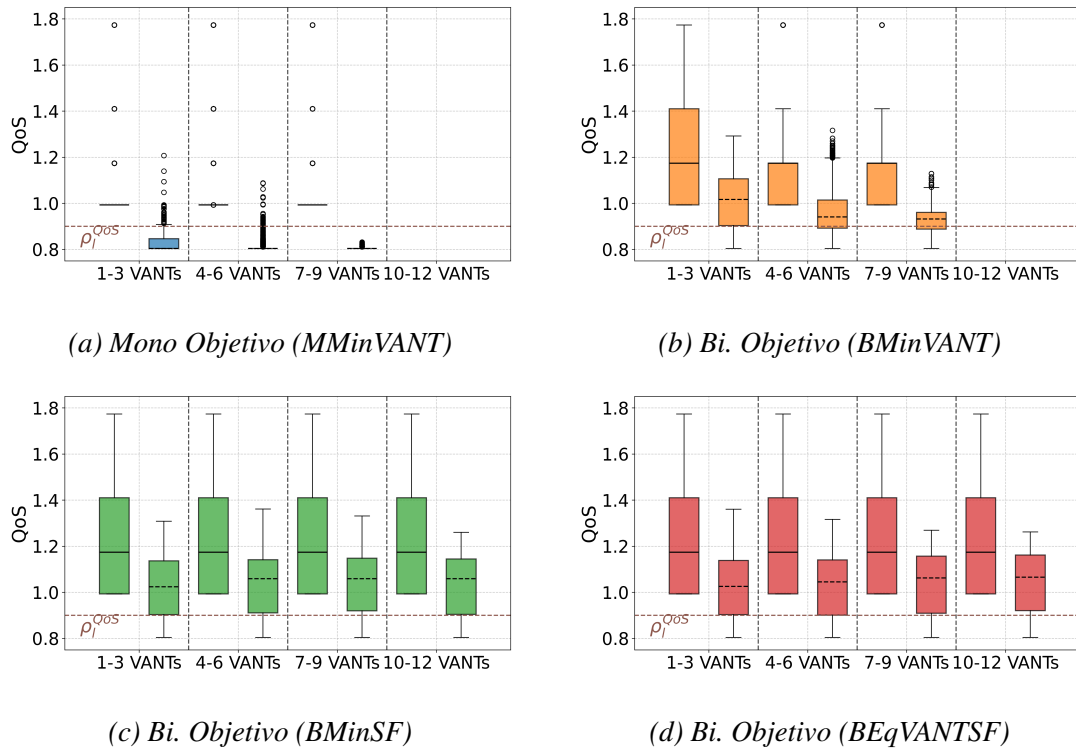


Figura 4.12: Distribuição da QoS por faixas de VANTs implantados, considerando diferentes estratégias de otimização e simulação. [Criada pelo Autor.]

A estrutura dos *boxplots* considera faixas de 1–3, 4–6, 7–9 e 10–12 VANTs, com cada par de gráficos representando os valores de otimização (esquerda) e simulação (direita) para cada estratégia.

A estratégia mono-objetivo **MMinVANT** apresenta os piores resultados simulados entre todas as abordagens (Figura 4.12a). Observa-se que, embora a otimização forneça valores razoáveis de QoS, as simulações revelam desempenho significativamente inferior, com valores concentrados abaixo de 1,0 e a maioria dos *outliers* posicionados abaixo de ρ_l^{QoS} . Em faixas superiores de VANTs, a variabilidade aumenta e a QoS permanece baixa, indicando que a estratégia não se sustenta frente aos efeitos reais de propa-

gação e interferência. O modelo apresenta elevada sensibilidade e baixa robustez quando exposto a condições práticas.

A abordagem biobjetivo **BMinVANT** (Figura 4.12b) exibe desempenho intermediário. As simulações demonstram valores medianos próximos a 1,0 nas faixas iniciais e ligeiramente superiores nas faixas intermediárias, com presença de *outliers* positivos. A dispersão dos valores simulados é visivelmente menor em comparação ao **MMinVANT**, indicando um modelo mais estável e menos sensível a variações externas. A estratégia consegue mitigar parte das limitações práticas e representa uma alternativa viável em cenários onde se busca reduzir o número de **VANTs** com impacto controlado na **QoS**.

A estratégia biobjetivo **BMinSF** (Figura 4.12c) apresenta comportamento robusto. As otimizações atingem valores medianos elevados (acima de 1,2), e as simulações mantêm distribuições consistentes, com a maioria dos valores acima de ρ_l^{QoS} . Há redução de dispersão nas faixas mais altas, sugerindo que o modelo se adapta bem à alocação balanceada de dispositivos entre os **SFs**, mesmo com o aumento da complexidade da rede. A presença de *outliers* acima da mediana reforça o bom desempenho em determinados cenários específicos.

Por fim, a abordagem **BEqVANTSf** (Figura 4.12d) demonstra a melhor consistência entre os valores teóricos e práticos. Os resultados de otimização mantêm medianas próximas de 1,2 a 1,4, enquanto as simulações apresentam menor dispersão e valores concentrados acima do limiar ρ_l^{QoS} . A ausência de *outliers* abaixo do primeiro quartil na maioria das faixas revela a confiabilidade do modelo frente a variações estocásticas. A estratégia mostra-se adequada para aplicações que requerem manutenção da qualidade de serviço com estabilidade.

Conclui-se portanto que, a estratégia **MMinVANT** não oferece desempenho satisfatório em ambientes reais, sendo altamente suscetível à degradação da **QoS**, a estratégia **BMinVANT** representa um bom compromisso entre custo e desempenho, especialmente para faixas intermediárias de **VANTs**, e as estratégias **BMinSF** e **BEqVANTSf** apresentam os melhores resultados em termos de robustez e qualidade, mantendo a **QoS** acima do limiar mínimo, mesmo nas simulações. A escolha entre as estratégias deve ser orientada pelos objetivos do operador, i.e., redução de custo (**CAPEX/OPEX**) com desempenho aceitável favorece **BMinVANT**, e por outro lado, alta confiabilidade e previsibilidade em cenários críticos demandam **BMinSF** ou **BEqVANTSf**.

Análise da Relação entre o Número de Dispositivos e o Número de VANTs Implantados

Esta subseção examina a relação entre a quantidade de dispositivos **LoRa-ED** na rede e o número médio de **VANTs** implantados, conforme definido pelas diferentes

estratégias de otimização. A Figura 4.13 apresenta a evolução desse número em função do crescimento da densidade de dispositivos.

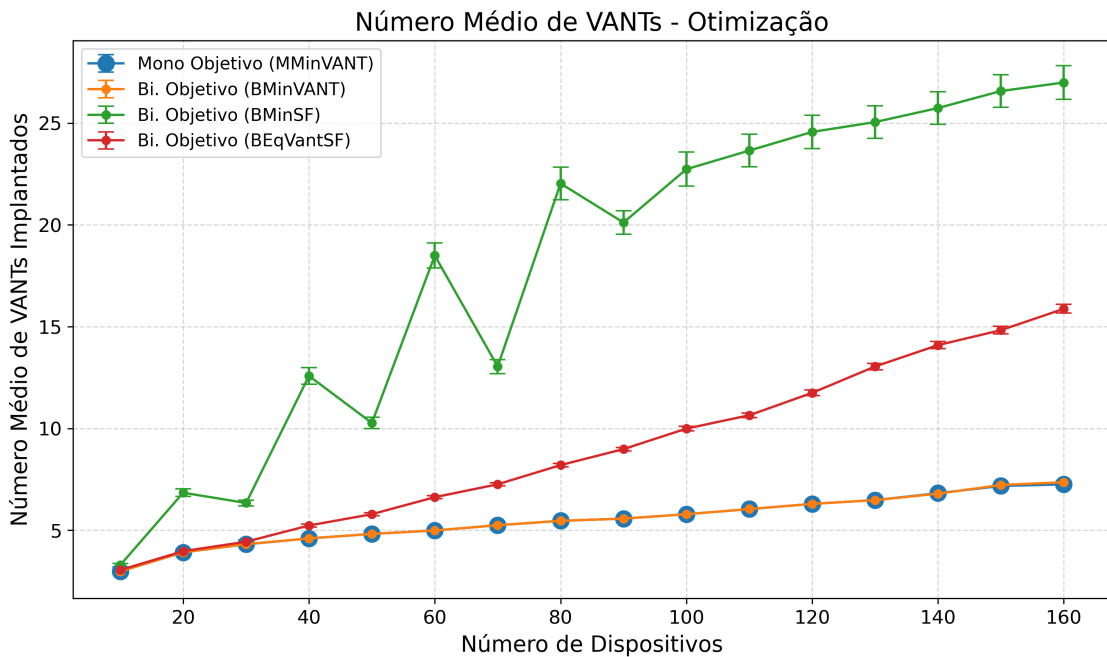


Figura 4.13: Número médio de VANTs implantados em função do número de dispositivos para as diferentes estratégias na otimização. [Criada pelo Autor.]

Os resultados evidenciam diferenças significativas entre as abordagens mono e biobjetivo no que diz respeito à alocação de infraestrutura aérea. As estratégias **BMinSF** e **BEqVANTSF** apresentam crescimento progressivo no número médio de VANTs implantados com o aumento da densidade de dispositivos, o que é esperado, dado que ambas priorizam a eficiência espectral e a distribuição equilibrada entre os fatores de espalhamento.

Notadamente, a estratégia **BMinSF** exibe um padrão de variação irregular, observam-se saltos e quedas pontuais, especialmente nas faixas entre 30 e 80 dispositivos. Essa oscilação pode ser atribuída a efeitos combinatórios da formulação biobjetivo, que, ao priorizar a minimização da concentração de dispositivos em um mesmo SF, força realocações conforme ocorram pequenos incrementos no número de dispositivos, comprometendo a homogeneidade. Assim, ao invés de uma adaptação gradual, o modelo responde com alterações no número de VANTs para garantir a manutenção dos níveis de QoS e da distribuição inter-SF.

Por sua vez, a estratégia **BEqVANTSF**, que também adota critérios de balanceamento espectral, exibe crescimento mais estável e contínuo, com menor dispersão. Isso evidencia sua capacidade de manter um equilíbrio entre cobertura e eficiência, sem com-

prometer de forma abrupta a alocação de recursos, tornando-se uma alternativa viável para cenários com maior previsibilidade e controle orçamentário.

A estratégia **BMinVANT**, cujo foco é a minimização do número de **VANTs**, mantém um crescimento linear e contido no número de unidades implantadas, mesmo em cenários densos. Seu desempenho é consistente, com baixos desvios, o que a torna ideal para ambientes com forte restrição de **CAPEX/OPEX**.

Finalmente, a abordagem mono-objetivo **MMinVANT** exibe os menores valores médios de **VANTs** em todas as faixas de dispositivos, alinhando-se perfeitamente com seu objetivo de reduzir a infraestrutura necessária. No entanto, análises anteriores demonstraram que tal economia compromete significativamente a **QoS**, especialmente sob aumento da carga de tráfego, o que limita sua aplicação em cenários críticos.

Cabe ressaltar que, na simulação subsequente, a quantidade de **VANTs** corresponde exatamente à configuração definida na otimização. Essa abordagem garante consistência na avaliação dos impactos reais de propagação e interferência sobre a alocação otimizada, permitindo validar a robustez das estratégias sob condições reais de operação.

Conclui-se, portanto, que o comportamento do número de **VANTs** implantados está diretamente relacionado ao critério de otimização priorizado. Estratégias como **BMinSF** maximizam a eficiência da alocação espectral, mesmo ao custo de oscilações abruptas na infraestrutura. Já abordagens como **BEqVANTSF** e **BMinVANT** demonstram comportamento mais linear e estável, com benefícios claros em termos de escalabilidade e previsibilidade. A escolha ideal dependerá do perfil de uso da rede, dos requisitos de **QoS** e das limitações orçamentárias do operador.

Análise de Interferências

Esta subseção analisa o impacto das diferentes estratégias de otimização nos níveis de interferência média observados na rede, considerando variáveis como o número de posições candidatas e a densidade de dispositivos.

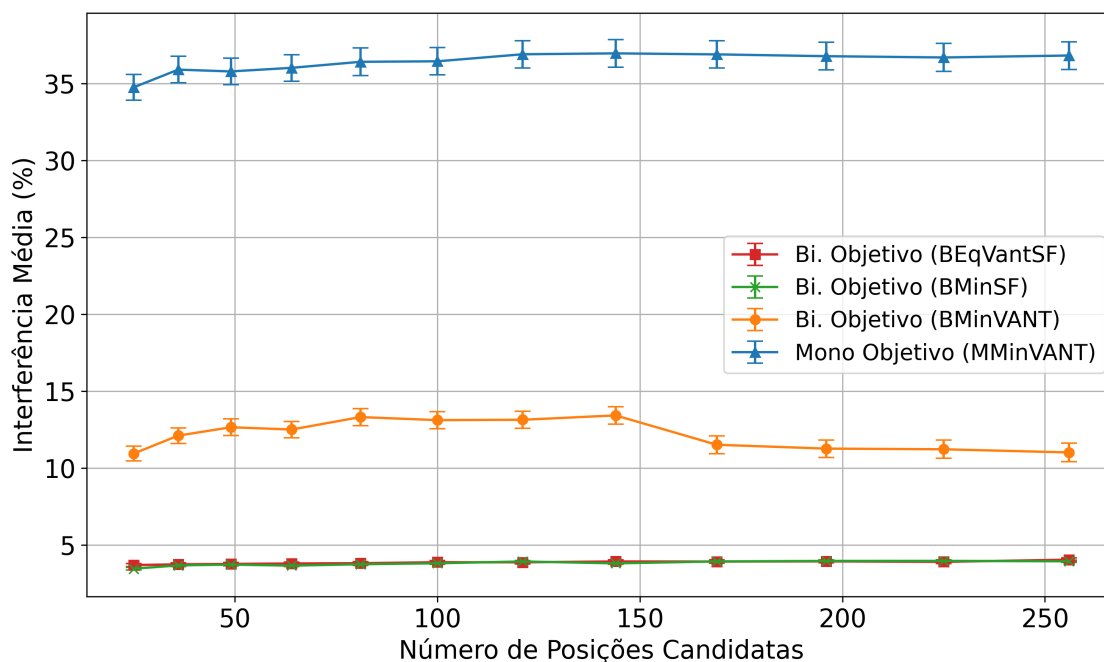


Figura 4.14: Interferência média em função do número de posições candidatas. [Criada pelo Autor.]

A Figura 4.14 apresenta a interferência média em função do número de posições candidatas disponíveis para a implantação dos VANTs. Observa-se que a estratégia monoobjetivo **MMinVANT**, voltada exclusivamente para a minimização da quantidade de VANTs, apresenta os maiores índices de interferência, com média próxima de 36% ao longo de todo o intervalo analisado. Esse comportamento decorre da elevada concentração de dispositivos em poucos enlaces, resultado da limitação do modelo ao desconsiderar aspectos de balanceamento espectral e diversidade de cobertura. A negligência de critérios relacionados à qualidade de enlace leva à sobreposição de transmissões com o mesmo SF, intensificando os níveis de colisão e degradação da comunicação.

Em contraste, as estratégias biobjetivo, em especial **BMinSF** e **BEqVANTSF**, demonstram desempenho significativamente superior quanto à mitigação de interferências. Ambas mantêm a interferência média abaixo de 5% ao longo de toda a análise, estabilizando-se em torno de 3,5% e 4%, respectivamente. Esses resultados evidenciam a efetividade das abordagens que integram critérios de distribuição de dispositivos entre os diferentes SFs com a minimização conjunta da infraestrutura, promovendo alocações mais eficientes e robustas. A estratégia **BMinVANT**, por sua vez, apresenta comportamento intermediário, com interferência oscilando entre 11% e 14%, sugerindo que a priorização da redução da infraestrutura, mesmo sob formulação biobjetivo, ainda impõe limitações no controle de interferência.

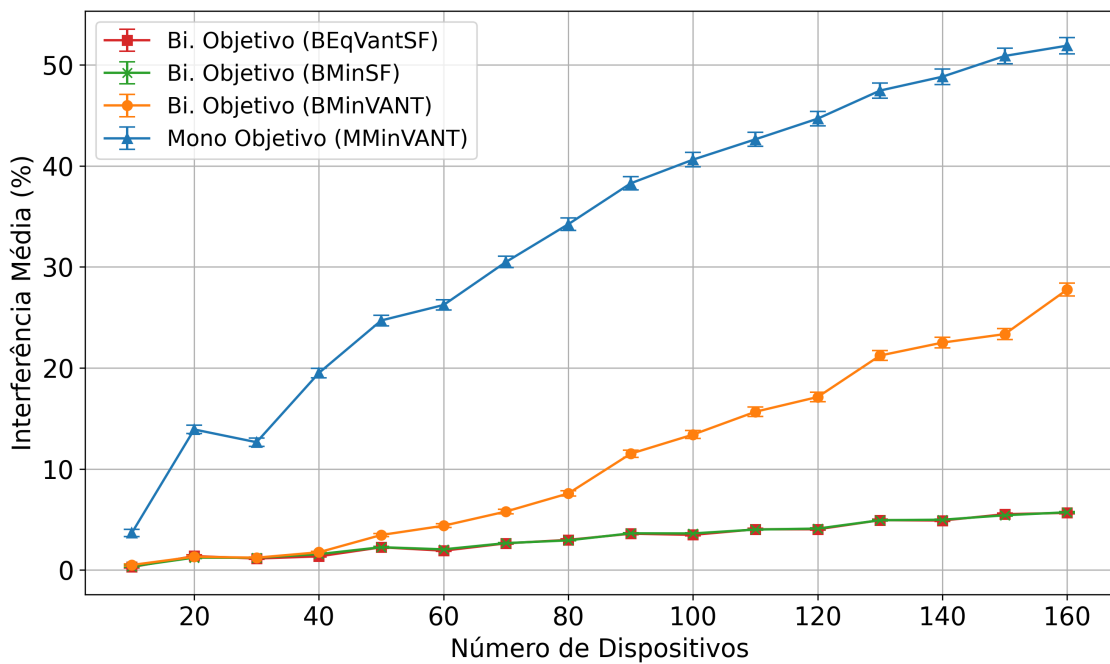


Figura 4.15: Interferência média em função do número de dispositivos. [Criada pelo Autor.]

A Figura 4.15 complementa a análise ao representar a interferência média em função da densidade de dispositivos na rede. Consta-se que a estratégia **MMinVANT** é particularmente sensível à densificação, apresentando crescimento acentuado nos níveis de interferência, que ultrapassam 52% para cenários com 160 dispositivos. Tal crescimento reflete a incapacidade da abordagem mono-objetivo de redistribuir eficientemente os dispositivos à medida que a carga da rede aumenta, acentuando a competição pelo meio e a sobreposição de sinais.

Em contrapartida, as estratégias **BMinSF** e **BEqVANTSF** mantêm interferência abaixo de 6% mesmo em cenários altamente densos, demonstrando maior estabilidade e resiliência à variação na carga da rede. A presença de múltiplos objetivos na formulação favorece alocações mais distribuídas e menos suscetíveis à saturação dos canais, reduzindo a probabilidade de colisões e perdas de pacotes. A estratégia **BMinVANT** novamente apresenta desempenho intermediário, embora demonstre aumento gradual da interferência com a densidade, mantendo-se entre 10% e 15%.

De modo geral, os resultados indicam que a inclusão simultânea de múltiplos critérios na formulação do problema permite uma alocação mais eficiente e adaptativa dos **VANTs**, resultando em menores níveis de interferência e maior confiabilidade da rede. As estratégias biobjetivo demonstraram-se mais eficazes para cenários de alta densidade, proporcionando melhor desempenho global, mesmo sob restrições topológicas severas.

Portanto, conclui-se que a formulação biobjetivo, ao equilibrar requisitos de cobertura, distribuição espectral e minimização de infraestrutura, constitui a abordagem

mais adequada para aplicações críticas em redes IoT heterogêneas e dinâmicas, garantindo melhor relação entre QoS e robustez operacional.

Discussão Geral dos Resultados

Os resultados obtidos evidenciam um compromisso inevitável entre a minimização do número de VANTs e a manutenção da qualidade de serviço (QoS) nas comunicações. Estratégias como BMinSF e BEqVANTSF destacaram-se pelo elevado desempenho em métricas de rede, promovendo uma alocação balanceada dos dispositivos LoRa-ED entre os diferentes SFs e reduzindo significativamente os índices de interferência. Contudo, tais ganhos operacionais demandam a implantação de um número maior de VANTs, o que implica aumento nos custos CAPEX e OPEX.

Por outro lado, a estratégia BMinVANT, centrada na minimização da infraestrutura, apresentou desempenho satisfatório em cenários de baixa e média densidade, conseguindo manter a QoS dentro dos limites exigidos. Tal característica a torna uma opção atrativa para aplicações que demandam eficiência econômica e possuem maior flexibilidade quanto às exigências de QoS.

A abordagem mono-objetivo MMinVANT, embora eficiente em reduzir a quantidade de VANTs, demonstrou severa degradação da QoS, especialmente sob altas densidades de dispositivos. Esse comportamento resulta da ausência de mecanismos de balanceamento de carga espectral, o que gera elevada concentração de dispositivos em poucos SFs, intensificando colisões e perda de pacotes.

Outro fator relevante observado foi a influência da granularidade espacial sobre os resultados. O aumento no número de posições candidatas favorece a diversidade de alocação e melhora a cobertura, mas também eleva a complexidade computacional do problema, exigindo maior capacidade de processamento por parte do solucionador. Identificou-se, entretanto, um ponto de saturação, além do qual os ganhos em QoS tornam-se marginais, indicando a existência de um ponto ótimo de resolução espacial.

A comparação entre os resultados otimizados e os simulados no ns-3 evidenciou variações expressivas, principalmente nas abordagens mais agressivas em termos de minimização. As simulações revelaram o impacto de fatores não modelados diretamente na formulação matemática, como interferência entre enlaces, perdas de propagação e variações estocásticas no ambiente de comunicação. Esses achados reforçam a necessidade de considerar soluções híbridas, que combinem a exatidão da otimização matemática com mecanismos adaptativos capazes de realizar ajustes em tempo de execução, garantindo maior robustez frente às condições reais da rede.

4.6 Conclusão

Neste capítulo, foi conduzida uma avaliação aprofundada das estratégias de alocação dinâmica de recursos baseadas em programação linear inteira mista (**MILP**), com foco na otimização da infraestrutura de comunicação em redes **IoT** não-**3GPP** mediadas por veículos aéreos não tripulados (**VANTs**). A formulação proposta considerou diferentes critérios de otimização, incluindo minimização do número de **VANTs**, distribuição balanceada entre **SFs**, restrições de capacidade de tráfego e requisitos de **QoS**.

Os resultados obtidos demonstraram que estratégias biobjetivo, especialmente **BMinSF** e **BEqVANTSF**, apresentam maior capacidade de garantir níveis elevados de desempenho da rede, ainda que ao custo de maior número de **VANTs** implantados. Em contrapartida, a estratégia **BMinVANT** revelou-se uma alternativa eficiente para cenários com restrições orçamentárias, promovendo um bom equilíbrio entre custo e qualidade de serviço. A abordagem mono-objetivo **MMinVANT**, apesar de alcançar a mínima infraestrutura, mostrou-se limitada em sua capacidade de manter os níveis de **QoS**, especialmente em cenários de alta densidade de dispositivos.

A integração com simulações no **ns-3** permitiu validar as soluções em condições próximas à operação real, revelando discrepâncias entre os valores otimizados e os resultados observados em campo. Essa constatação destaca a relevância da adoção de estratégias adaptativas e sensíveis às condições dinâmicas da rede, especialmente em ambientes urbanos complexos e heterogêneos.

Dessa forma, este capítulo consolidou a viabilidade e os limites das soluções exatas baseadas em **MILP** para o problema de alocação dinâmica de recursos. No Capítulo 5, será introduzida uma nova abordagem baseada em **DRL**, com o objetivo de ampliar a flexibilidade, a escalabilidade e a adaptabilidade do sistema, superando as limitações das técnicas exatas ao incorporar inteligência adaptativa às decisões de posicionamento e alocação dos **VANTs**.

Abordagem Baseada em Aprendizado de Máquina

O Capítulo 5 apresenta uma abordagem baseada em **DRL** para a alocação dinâmica de recursos em redes **IoT** assistidas por **VANTs**. Nesta proposta, são exploradas arquiteturas adaptativas capazes de otimizar a tomada de decisões em tempo de execução, considerando as dinâmicas de mobilidade, interferência e heterogeneidade inerentes a redes não-**3GPP**.

O problema é formulado como um **MDP**, permitindo que o agente de aprendizado interaja com o ambiente, receba recompensas e aprimore sua política de ação com base nas observações extraídas da rede. Para resolver tal problema, este capítulo propõe e implementa quatro variantes de algoritmos **DRL**, e.g., **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**. Tais variantes foram desenvolvidas a partir das formulações originais dos respectivos algoritmos, com modificações específicas para acomodar as particularidades da alocação de recursos em redes **LoRaWAN-5G** integradas por **VANTs**.

Adicionalmente, este capítulo descreve o processo de modelagem dos estados, ações e recompensas no contexto do **MDP**, bem como as técnicas de engenharia de rede neural utilizadas na construção das arquiteturas de decisão. Os métodos de treinamento adotados incluem técnicas de exploração-exploração, normalização de recompensas, e otimização com base em *experience replay* e algoritmos de gradiente de política.

A avaliação do desempenho dos modelos propostos é conduzida em um ambiente de simulação construído com base no simulador **ns-3**, configurado para refletir cenários urbanos realistas, conforme estabelecido no Capítulo 4. São analisadas métricas como taxa de entrega de pacotes, **QoS**, interferência, tempo de convergência e estabilidade das decisões. Os resultados obtidos são comparados às soluções exatas baseadas em **MILP**, destacando os ganhos em adaptabilidade, escalabilidade e robustez dos métodos com **DRL**.

Este capítulo está estruturado da seguinte forma: a Seção 5.1 apresenta a modelagem formal do problema como um **MDP**, a Seção 5.2 descreve as arquiteturas dos algoritmos de aprendizado de máquina utilizados, a Seção 5.3 detalha o processo de trei-

namento e os parâmetros de simulação, a Seção 5.4 analisa os resultados obtidos, e, por fim, a Seção 5.5 discute as conclusões e implicações dos resultados experimentais.

5.1 Modelagem MDP

O paradigma fundamental de um processo de decisão de Markov (MDP) baseia-se na interação entre um *agente* e um *ambiente*, por meio da execução de ações. Nesse contexto, o problema de decisão sequencial é estruturado como um conjunto de *estados*, nos quais cada estado representa uma configuração específica do sistema. Uma premissa central no modelo MDP é a propriedade *Markoviana*, que estabelece que a distribuição de probabilidades dos estados futuros depende apenas do estado atual e da ação executada, sendo condicionalmente independente dos estados passados, dado o conhecimento perfeito do presente [Mahadevan 2009].

A estruturação em um modelo Markoviano simplifica a previsão de estados futuros ao considerar apenas o estado atual e a ação escolhida, reduzindo a complexidade computacional e permitindo escalabilidade para problemas maiores [Hamdani, Fergani e Ramdani 2021]. Essa propriedade viabiliza o uso de algoritmos de RL, como o QL e o DQN, para otimizar políticas de decisão e maximizar recompensas ao longo do tempo. Embora esses algoritmos sejam inspirados na equação de Bellman, um conceito central da programação dinâmica, eles operam em ambientes onde o modelo é desconhecido (*model-free*), aprendendo diretamente a partir de interações com o ambiente [Feriani e Hossain 2021]. Além disso, o modelo estabelece relações claras entre ações, estados e recompensas, ampliando sua aplicabilidade em diversos contextos de otimização.

Dessa forma, o MDP estabelece uma estrutura formal para tomada de decisão sequencial, composta por cinco elementos principais, definidos como uma tupla $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$, onde:

- \mathcal{S} representa o conjunto de estados possíveis;
- \mathcal{A} é o espaço de ações disponíveis;
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ é a função de transição entre estados;
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ é a função de recompensa associada a cada par estado-ação;
- $\gamma \in [0, 1]$ é o fator de desconto aplicado às recompensas futuras.

Neste trabalho, o espaço de estados \mathcal{S} é definido pelas combinações possíveis de posicionamento dos VANTs sobre um conjunto discreto de posições candidatas \mathcal{P} . Considerando-se que o número de VANTs a serem posicionados (δ) devem ser implantados em um subconjunto de $|\mathcal{P}|$ posições, o número total de estados possíveis é determi-

nado pela combinação sem repetição de δ elementos de \mathcal{P} , conforme

$$|\mathcal{S}| = C_{(|\mathcal{P}|, \delta)} = \frac{|\mathcal{P}|!}{\delta! \cdot (|\mathcal{P}| - \delta)!}. \quad (5-1)$$

A Figura 5.1 ilustra o crescimento exponencial do espaço de estados em função de δ e da cardinalidade do conjunto de posições candidatas $|\mathcal{P}|$.

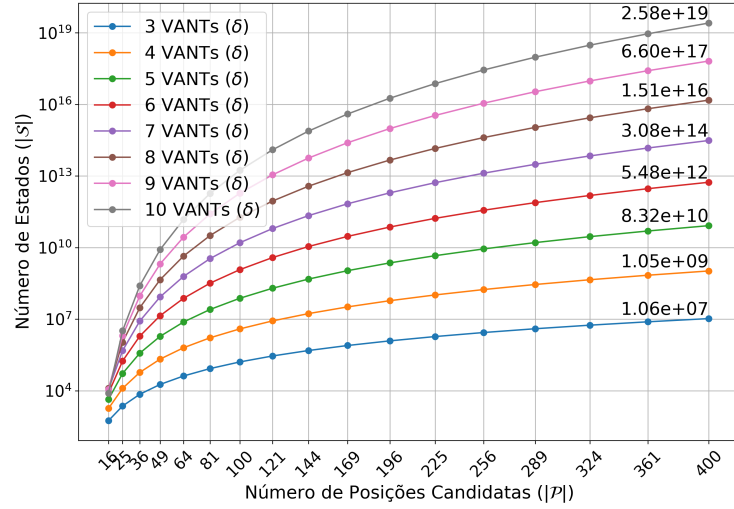


Figura 5.1: Crescimento do espaço de estados $|\mathcal{S}|$ em função do número de posições candidatas $|\mathcal{P}|$ e do número de VANTs δ . [Criada pelo Autor.]

Observa-se que o crescimento do espaço de estados é exponencial com o aumento de δ e $|\mathcal{P}|$. Esse crescimento exponencial representa um grande desafio para algoritmos de RL, pois aumenta significativamente o número de combinações possíveis que o agente precisa considerar. Com um maior espaço de estados, o treinamento se torna mais complexo e requer maior capacidade computacional e estratégias eficientes para explorar o ambiente de forma eficaz. Este contexto ressalta a importância da adoção de técnicas de DRL, que são capazes de lidar com grandes espaços de estados através do uso de aproximações por redes neurais.

O conjunto das possíveis ações foi definido como

$$\mathcal{M} = \{m_{norte}, m_{sul}, m_{leste}, m_{oeste}, m_{\otimes}\}, \quad (5-2)$$

onde, m_{norte} indica a ação de mover o VANT para a direção cardinal norte, m_{sul} indica mover o VANT para a direção cardinal sul, e assim respectivamente para as outras direções cardinais e m_{\otimes} indica a decisão por manter o VANT parado na mesma posição. Desta forma, o espaço de ações é definido pelo total de possíveis arranjos com repetições $\mathcal{A} = \{a_1, a_2, \dots, a_{|\mathcal{A}|}\}$, e $|\mathcal{A}| = |\mathcal{M}|^\delta$, e.g., se $\delta = 2$, então \mathcal{A} conterà as ações

$\{(m_{norte}, m_{norte}), (m_{norte}, m_{sul}), \dots, (m_{\otimes}, m_{\otimes})\}$, totalizando $5^2 = 25$ ações. Assim, cada ação será uma tupla, indicando o movimento de cada VANT a ser posicionado. A Figura 5.2 demonstra o crescimento exponencial do espaço de ações em função do número de VANTs.

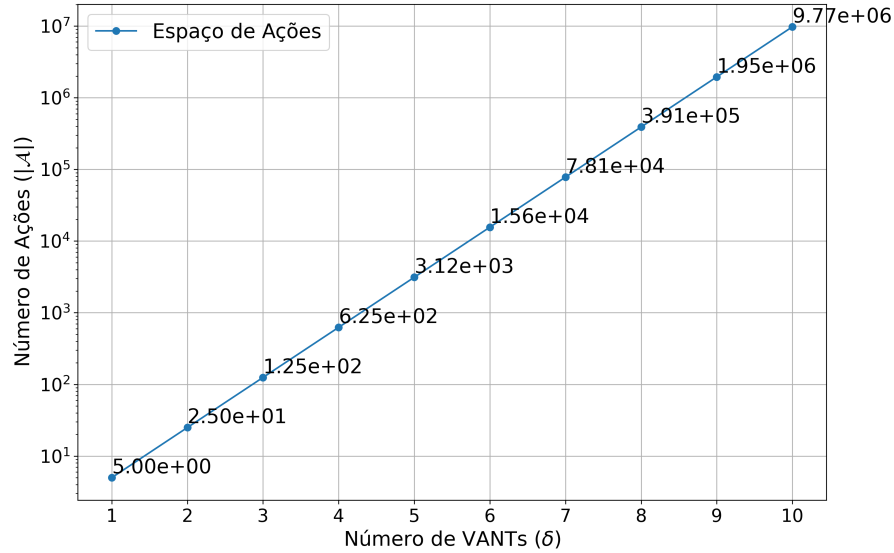


Figura 5.2: Gráfico do crescimento de $|A|$ em função de δ . [Criada pelo Autor.]

Nota-se que o aumento do número de VANTs provoca um crescimento exponencial no espaço de ações. Esse crescimento implica em um aumento substancial na complexidade computacional para encontrar políticas ótimas, uma vez que o agente precisa explorar e aprender sobre um conjunto significativamente maior de possibilidades de movimentação. Em problemas de alta dimensionalidade, como este, a complexidade pode impactar diretamente o tempo de convergência do aprendizado e a eficiência das decisões tomadas [Sutton e Barto 2018]. Para mitigar esse impacto, a utilização de algoritmos avançados de RL torna-se essencial, visto que tais abordagens são mais robustas para lidar com grandes espaços de ações e permitem maior eficiência na busca por soluções ótimas.

A função de transição representa a probabilidade de mudar de um estado s para o estado s' por meio da ação a e é definida como $\mathcal{T}(s'|s, a) = \varphi(s'|s, a)$, onde $\varphi(s'|s, a)$ 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 atribuída a cada transição é definida a partir da média de QoS observada no estado s , conforme

$$\mu_{QoS_s} = \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} QoS_{k,s}, \quad (5-3)$$

onde μQoS_s representa a média da qualidade de serviço e ρ_l^{QoS} é o limiar mínimo exigido de QoS para o *slice* l .

A função de recompensa é composta por dois componentes, a recompensa baseada na média de qualidade de serviço observada no estado de destino, e uma penalidade associada à ocorrência de eventos indesejados durante a transição.

A recompensa primária $\xi(s, a, s')$ é definida como:

$$\xi(s, a, s') = \begin{cases} +\mu QoS_{s'} & \text{se } \mu QoS_{s'} > \rho_l^{QoS}, \\ -\mu QoS_{s'} & \text{se } \mu QoS_{s'} < \rho_l^{QoS}, \\ 0 & \text{se } \mu QoS_{s'} = \rho_l^{QoS}, \end{cases} \quad (5-4)$$

onde $\mu QoS_{s'}$ representa a média de QoS no estado s' e ρ_l^{QoS} é o limiar mínimo de QoS definido para o *slice* l .

A função $\zeta(s, a, s')$ representa a penalidade associada a eventos indesejados durante a transição (s, a, s') :

$$\zeta(s, a, s') = \begin{cases} -1 & \text{se houver saída da área delimitada,} \\ -2 & \text{se ocorrer colisão entre VANTs,} \\ 0 & \text{caso contrário.} \end{cases} \quad (5-5)$$

Assim, a recompensa total atribuída ao agente ao transitar de s para s' por meio da ação a é definida por

$$\mathcal{R}(s, a, s') = \xi(s, a, s') + \zeta(s, a, s'). \quad (5-6)$$

A política π é uma função que define o comportamento do agente ao mapear estados para ações, ou seja, $\pi: \mathcal{S} \rightarrow \mathcal{A}$. Essa função determina qual ação o agente deve executar em cada estado $s \in \mathcal{S}$ com o objetivo de maximizar as recompensas acumuladas ao longo do tempo.

O valor de uma política π em um estado s , denotado por $V^\pi(s)$, é definido como a expectativa da soma das recompensas futuras esperadas ao seguir a política π a partir do estado s , conforme

$$V^\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t, s_{t+1}) \mid s_0 = s \right], \quad (5-7)$$

onde $a_t = \pi(s_t)$ e $s_{t+1} \sim \mathcal{T}(s_{t+1} | s_t, a_t)$, sendo $\gamma \in [0, 1]$ o fator de desconto aplicado às recompensas futuras.

A política ótima π^* é aquela que maximiza o valor esperado para todos os estados

do espaço \mathcal{S} . Assim, a definição da política ótima pode ser expressa como

$$\pi^* = \arg \max_{\pi} V^{\pi}(s), \quad \forall s \in \mathcal{S}. \quad (5-8)$$

O valor ótimo de uma ação a em um estado s ao seguir a política ótima π^* é representado pela função *Q-valor ótima*, denotada por $Q^*(s, a)$. Esta função estima a recompensa esperada ao executar a ação a no estado s e, a partir daí, seguir a política ótima. Sua definição é dada por

$$Q^*(s, a) = \mathcal{R}(s, a, s') + \gamma \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \cdot V^*(s'). \quad (5-9)$$

Dessa forma, a política ótima também pode ser expressa em termos da função Q-valor como

$$\pi^*(s) = \arg \max_{a \in \mathcal{A}} Q^*(s, a), \quad (5-10)$$

onde o agente escolhe, em cada estado s , a ação que maximiza o valor esperado de recompensa acumulada.

A Tabela 5.1 resume os símbolos adotados para o MDP.

Tabela 5.1: Lista de símbolos adotados na descrição do MDP, organizados por categoria. [Criada pelo Autor.]

	Notação	Descrição
Conjuntos	\mathcal{S}	Conjunto de estados possíveis do ambiente
	\mathcal{A}	Conjunto de ações disponíveis para o agente
	\mathcal{K}	Conjunto de dispositivos LoRa-ED
	\mathcal{P}	Conjunto de posições candidatas para alocação dos VANTs
Funções	$\mathcal{T}(s' s, a)$	Função de transição para s' a partir de s ao executar a
	$\mathcal{R}(s, a, s')$	Função de recompensa total atribuída à transição (s, a, s')
	$\xi(s, a, s')$	Recompensa associada à média de QoS no estado s'
	$\zeta(s, a, s')$	Penalidade atribuída a eventos como colisões ou saídas da área
	$\pi(s)$	Política que mapeia o estado s para uma ação $a \in \mathcal{A}$
	$\mathcal{V}^{\pi}(s)$	Valor esperado acumulado ao seguir a política π a partir do estado s
	$Q(s, a)$	Valor esperado de executar a ação a no estado s e seguir a política π
	$Q^*(s, a)$	Função de valor ótima associada à política ótima π^*
	$\pi^*(s)$	Política ótima que maximiza o valor esperado em cada estado s
Parâmetros	μ_{QoS_s}	Média da QoS dos dispositivos no estado s
	ρ_l^{QoS}	Limiar mínimo de QoS exigido para o <i>slice</i> l
	δ	Número de VANTs a serem implantados
	γ	Fator de desconto aplicado às recompensas futuras ($\gamma \in [0, 1]$)

5.2 Formulação do Problema

Nesta seção, é apresentada a formulação matemática do problema de alocação dinâmica de recursos em redes IoT não-3GPP com suporte de VANTs. O objetivo principal consiste em definir uma estratégia adaptativa e eficiente para otimizar, de forma contínua, o posicionamento dos VANTs e a alocação dos recursos de comunicação, com vistas à maximização de métricas de desempenho como latência, vazão e QoS.

Diferentemente das abordagens exatas baseadas em otimização combinatória, aqui adota-se uma perspectiva de aprendizado baseada em interação, por meio de algoritmos de DRL. Nessa abordagem, os VANTs são modelados como agentes inteligentes capazes de explorar o ambiente, tomar decisões sequenciais e aprender políticas ótimas de posicionamento e alocação com base em interações simuladas.

O ambiente utilizado para treinamento é modelado no simulador ns-3, configurado para refletir cenários urbanos realistas e dinâmicos. A cada interação, o agente observa o estado da rede, seleciona uma ação e recebe uma recompensa com base no impacto dessa decisão sobre o desempenho global do sistema. Essa estrutura possibilita a adaptação contínua às variações do ambiente e ao posicionamento dos dispositivos LoRa-ED, bem como à presença de interferências e restrições topológicas.

As subseções seguintes detalham os principais algoritmos de DRL empregados neste trabalho, descrevendo suas estruturas, funções de valor, estratégias de política, técnicas de treinamento e adaptações específicas introduzidas para lidar com as particularidades do problema de alocação dinâmica em redes heterogêneas assistidas por VANTs.

DRL para Alocação Dinâmica de Recursos em Redes IoT envolvendo VANTs

O aprendizado por reforço profundo tem se consolidado como uma abordagem eficaz para a otimização de redes IoT, especialmente em cenários desafiadores que envolvem mobilidade dinâmica, variabilidade topológica e restrições de comunicação [Tellache et al. 2022, Esteves et al. 2021, Wang et al. 2020]. Nesse contexto, agentes inteligentes, representados por VANTs equipados com gateways LoRaWAN, são capazes de interagir com o ambiente simulado no ns-3, aprendendo políticas de posicionamento e movimentação a partir das recompensas obtidas. Tais recompensas refletem a qualidade das decisões tomadas, possibilitando a convergência para políticas ótimas que maximizem o desempenho global da rede.

Para formalizar matematicamente o problema e orientar o processo de aprendizado, foi adotado uma modelagem MDP, conforme descrito na Seção 5.1. Nesse modelo, o espaço de estados \mathcal{S} é definido pelas possíveis combinações de posições dos VANTs sobre o conjunto de locais candidatos \mathcal{P} . O espaço de ações \mathcal{A} contempla as decisões de

movimentação de cada VANT, incluindo deslocamentos em múltiplas direções ou a permanência na posição atual. A função de recompensa $\mathcal{R}(s, a, s')$ foi projetada para refletir a qualidade de cada transição, atribuindo valores positivos para ações que aumentam a conectividade e a QoS, e penalidades para comportamentos indesejados, como colisões entre VANTs, saída da área operacional ou degradação na QoS.

A função de valor de ação $Q(s, a)$ desempenha um papel central no processo de decisão, pois estima o retorno esperado de executar a ação a no estado s e seguir a política π a partir desse ponto. Dado o elevado número de combinações de estados e ações, a função Q será aproximada por meio de redes neurais profundas (DNNs). Cada agente DRL considerado neste trabalho — SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO — adota uma arquitetura de rede neural adaptada às suas particularidades. Essas redes são treinadas para estimar os valores Q de forma eficiente, permitindo a aprendizagem de políticas robustas, mesmo em ambientes de alta dimensionalidade.

As arquiteturas das DNNs foram ajustadas para refletir as características específicas de cada agente, incluindo a definição da estrutura de entrada, que representa o estado atual da rede, e do vetor de saída, que corresponde aos valores estimados para as ações possíveis. Técnicas consolidadas na literatura, como *target networks*, *experience replay* e funções de perda específicas, foram incorporadas para garantir estabilidade e eficiência no processo de treinamento.

Com essa base teórica e computacional estabelecida, as seções subsequentes apresentam os algoritmos de DRL desenvolvidos, destacando suas principais características, adaptações específicas ao problema de alocação dinâmica de recursos em redes IoT assistidas por VANTs e os resultados obtidos em ambiente simulado.

Shaped-Reward Deep Q-Network

O algoritmo *Shaped-Reward Deep Q-Network* (SR-DQN) consiste em uma adaptação do algoritmo clássico DQN para o contexto de alocação dinâmica de recursos em redes de comunicação IoT com suporte aéreo fornecido por VANTs. A proposta foi desenvolvida com o objetivo de permitir que os VANTs aprendam de forma autônoma políticas de posicionamento que maximizem a QoS. O Algoritmo 5.1 descreve o funcionamento completo do SR-DQN, incluindo os procedimentos de inicialização, exploração ϵ -greedy, armazenamento de experiências, atualização dos pesos da rede neural e sincronização periódica da rede alvo.

No algoritmo SR-DQN, a abordagem clássica do DQN é aprimorada por meio da técnica de *shaped-reward*, que consiste em moldar a função de recompensa para refletir com maior sensibilidade os objetivos do sistema e guiar o aprendizado do agente de forma mais eficaz. A função de recompensa é ajustada para incorporar penalizações e incentivos

baseados em conhecimento de domínio, como colisões entre VANTs, permanência dentro da área delimitada e manutenção dos requisitos mínimos de QoS para os dispositivos conectados. Essa modelagem detalhada das recompensas permite acelerar a convergência do treinamento, uma vez que o agente recebe sinais de aprendizado mais informativos mesmo em situações intermediárias, reduzindo a ocorrência de políticas subótimas e melhorando a exploração do espaço de estados.

Algoritmo 5.1: *Shaped-Reward Deep Q-Network (SR-DQN)* para Alocação Dinâmica de Recursos em Redes IoT com VANTs

Entrada: Ambiente ns-3, \mathcal{S} , \mathcal{A} , α , γ , ϵ , \mathcal{B} , \mathcal{C} , E_{max} , T_{max}

Saída: Política ótima π^* para alocação de VANTs

Inicialização:

Inicializar Rede- Q com pesos θ

Inicializar Target-network com pesos θ'

Inicializar Replay Buffer \mathcal{D}

Configurar e inicializar ambiente ns-3 e obter s_0

para $e \leftarrow 1$ **to** E_{max} **faça**

para $t \leftarrow 1$ **to** T_{max} **faça**

 Selecionar ação a_t com política ϵ -greedy

 Executar a_t e obter $(s', r, done)$

 Armazenar $(s, a, r, s', done)$ em \mathcal{D}

se $len(\mathcal{D}) > \mathcal{B}$ **então**

 Amostrar \mathcal{B} experiências $(s_j, a_j, r_j, s'_j, done_j)$

 Computar $y_j = r_j + \gamma \max_{a'} Q(s'_j, a'; \theta')$

 Minimizar $\mathcal{L}(\theta) = \frac{1}{\mathcal{B}} \sum_j (y_j - Q(s_j, a_j; \theta))^2$

 Atualizar pesos θ via Gradiente Descendente

 Atualizar estado $s \leftarrow s'$

se $done$ **então**

break

se $e \bmod \mathcal{C} = 0$ **então**

 Atualizar rede-alvo $\theta' \leftarrow \theta$

 Reiniciar ambiente e obter estado inicial s_0

Retorne Política aprendida π^*

O algoritmo apresenta particularidades que o distinguem do DQN clássico. Em primeiro lugar, ele foi concebido para operar de forma integrada ao ambiente de simulação ns-3, o que permite representar com fidelidade os cenários e dinâmicas de rede. Essa integração possibilita ao agente aprender uma política ótima de alocação de VANTs, considerando parâmetros realistas de propagação, interferência e tráfego. As recompensas $\mathcal{R}(s, a, s')$ e as penalidades associadas às ações são obtidas diretamente por meio da

interação com o [ns-3](#), que computa os valores a partir das condições da rede em cada estado, refletindo o impacto real das decisões sobre o desempenho da comunicação.

Embora o algoritmo utilize uma *target network* e um *replay buffer*, componentes tradicionalmente empregados em algoritmos [DQN](#) para estabilizar o processo de treinamento, essas estruturas foram adaptadas ao contexto específico do problema de alocação dinâmica de [VANTs](#). A atualização da rede-alvo é realizada por meio de um *hard update*, no qual os pesos da rede-alvo (θ') são completamente substituídos pelos pesos da rede principal (θ) após cada C episódios, seguindo a regra $\theta' \leftarrow \theta$. Essa abordagem mantém a rede-alvo constante entre atualizações sucessivas, contribuindo para a estabilidade do aprendizado e evitando oscilações abruptas nos valores estimados de Q .

Por contraste, o *soft update*, adotado em algumas variantes do [DQN](#), atualiza a rede-alvo de forma incremental segundo a equação $\theta' \leftarrow \tau\theta + (1 - \tau)\theta'$, onde $\tau \in (0, 1)$ define a taxa de atualização. Apesar de suavizar a transição entre as redes, o *soft update* impõe maior custo computacional e complexidade de controle. Diante disso, optou-se pela aplicação do *hard update*, cuja simplicidade operacional e eficiência computacional mostraram-se mais adequadas para integração com o simulador [ns-3](#). Nesse ambiente, onde as interações entre ações, estados e recompensas envolvem elevada complexidade e latência de resposta, o uso do *hard update* proporcionou maior robustez e estabilidade ao processo de aprendizado.

O algoritmo [SR-DQN](#) demonstra ser uma abordagem eficaz para problemas de aprendizado por reforço. No entanto, uma de suas principais limitações reside na superestimação dos valores Q , que pode levar a políticas subótimas devido ao uso do mesmo conjunto de pesos para selecionar e avaliar as ações. Essa deficiência motivou o desenvolvimento de técnicas mais avançadas, como o [DA-DDDQN](#). A Seção 5.2 explora os detalhes do [DA-DDDQN](#) e as técnicas para superar essas limitações.

Domain-Aware Double Dueling Deep Q-Network

Embora o algoritmo [SR-DQN](#) tenha se mostrado eficaz nos cenários de alocação dinâmica de [VANTs](#), ele apresenta uma limitação estrutural relevante, a tendência à superestimação dos valores Q durante o processo de aprendizado. Essa superestimação ocorre porque o mesmo conjunto de parâmetros da rede é utilizado tanto para a seleção quanto para a avaliação das ações, o que pode induzir o agente a adotar políticas subótimas, especialmente em ambientes com alta variabilidade e ruído nas recompensas. Para mitigar esse problema, propõe-se o Algoritmo 5.2 ([DA-DDDQN](#)). Essa abordagem combina três avanços importantes sobre o [DQN](#) original, e.g., o uso do mecanismo *double Q-learning*, a arquitetura *dueling*, e a incorporação de conhecimento do domínio (*domain-awareness*).

O *Double Q-Learning* reduz o viés de superestimação dos valores Q ao utilizar duas redes distintas, a rede de política, responsável por selecionar a ação, e a rede-alvo, que avalia o valor dessa ação. Por sua vez, a arquitetura *Dueling Network* separa o cálculo do valor do estado $V(s)$ e da vantagem da ação $A(s, a)$. O valor do estado representa a qualidade geral de estar em um determinado estado, independentemente da ação tomada, enquanto a vantagem mede o impacto específico de uma ação sobre o estado atual. Essa distinção permite que o agente identifique quais ações são mais relevantes em diferentes contextos, resultando em decisões mais precisas e aprendizado mais robusto. A característica *Domain-Aware* do algoritmo refere-se à incorporação explícita de conhecimento de domínio sobre a estrutura da rede e as restrições de mobilidade dos VANTs no ambiente de simulação ns-3, o que permite uma modelagem mais precisa das funções de recompensa e uma avaliação mais realista das transições de estado. Essa consciência de domínio contribui para uma política de decisão mais aderente às dinâmicas da rede IoT, melhorando a generalização e a convergência do modelo.

O Algoritmo 5.2 descreve o processo de treinamento do DA-DDDQN para a alocação dinâmica de recursos em redes IoT com VANTs. Inicialmente, são configuradas as redes neural de política e rede-alvo, com pesos θ e θ' , respectivamente, além do *Replay Buffer* \mathcal{D} e o ambiente de simulação ns-3. Em cada episódio de treinamento, o ambiente é resetado e o agente interage com o ambiente durante T_{max} passos, selecionando ações por meio de uma política ϵ -greedy baseada na Rede-Q. As experiências são armazenadas no *Replay Buffer*, e, quando o *buffer* atinge o tamanho mínimo \mathcal{B} , inicia-se o processo de aprendizado.

Utilizando a abordagem *Double Q-Learning*, o agente seleciona a melhor ação com a rede de política e avalia seu valor com a rede-alvo, calculando a recompensa esperada. Em seguida, a arquitetura *Dueling Network* é aplicada para estimar separadamente o valor do estado $V(s_j)$ e a vantagem da ação $A(s_j, a_j)$, combinando-os para calcular o valor $Q(s_j, a_j)$. A função de perda é então calculada e utilizada para atualizar os pesos da rede principal por meio de gradiente descendente. Periodicamente, a rede-alvo é sincronizada com a rede de política, e o parâmetro ϵ é decaído para equilibrar a exploração e a exploração estratégica ao longo dos episódios. O processo se repete até que a política π^* seja aprendida, garantindo uma estratégia eficiente de alocação de recursos no ambiente simulado.

Embora o DA-DDDQN apresente avanços significativos em termos de estabilidade e redução da superestimação dos valores Q , seu desempenho ainda depende de estratégias rigorosas de treinamento e da escolha adequada de hiperparâmetros para garantir a eficiência do aprendizado. Para superar algumas dessas limitações, especialmente quanto à dependência de estimativas precisas de valores de ação, propõe-se o algoritmo NSE-A2C. Este algoritmo adota uma abordagem baseada em políticas, utilizando os com-

ponentes **ator** e **crítico** para otimizar diretamente a política de decisão, proporcionando maior estabilidade e eficiência em ambientes complexos e de grande dimensionalidade.

Algoritmo 5.2: *Domain-Aware Double Dueling Deep Q-Network (DA-DDDQN)* para Alocação Dinâmica de Recursos em Redes IoT com VANT

Entrada: Ambiente **ns-3**, \mathcal{S} , \mathcal{A} , γ , ϵ_{\max} , ϵ_{\min} , ϵ_{decay} , α , \mathcal{B} , \mathcal{C} , E_{\max} , T_{\max}

Saída: Política treinada π^* para alocação de **VANTs**

Inicialização:

Inicializar Rede- Q e Target- Q com pesos θ e θ'

Inicializar *Replay Buffer* \mathcal{D}

Configurar e inicializar o ambiente **ns-3** e obter s_0

para $e \leftarrow 1$ **to** E_{\max} **faça**

para $t \leftarrow 1$ **to** T_{\max} **faça**

 Selecionar a ação a_t utilizando política ϵ -greedy baseada na Rede- Q

 Executar a_t e obter $(s', r, done)$

 Armazenar $(s, a, r, s', done)$ em \mathcal{D}

se $len(\mathcal{D}) > \mathcal{B}$ **então**

 Amostrar \mathcal{B} experiências $(s_j, a_j, r_j, s'_j, done_j)$

Double Q-Learning:

 Determinar $\hat{a} = \arg \max Q(s'_j, a'; \theta)$

 Estimar $y_j = r_j + \gamma \cdot (1 - done_j) \cdot Q(s'_j, \hat{a}; \theta')$

Arquitetura Dueling:

 Estimar o valor do estado $V(s_j)$ e a vantagem da ação $A(s_j, a_j)$

 Combinar para calcular $Q(s_j, a_j) = V(s_j) + \left(A(s_j, a_j) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s_j, a') \right)$

 Calcular a função de perda $\mathcal{L}(\theta) = \frac{1}{\mathcal{B}} \sum_j (y_j - Q(s_j, a_j; \theta))^2$

 Atualizar pesos θ via Gradiente Descendente

 Atualizar estado $s \leftarrow s'$

se $done$ **então**

break

se $e \bmod \mathcal{C} = 0$ **então**

 Atualizar a rede-alvo $\theta' \leftarrow \theta$

 Atualizar $\epsilon \leftarrow \max(\epsilon \cdot \epsilon_{decay}, \epsilon_{min})$

 Reinicializar o ambiente e obter estado inicial s_0

Retorne Política aprendida π^*

Normalized-State Entropy Advantage Actor-Critic

Algoritmos baseados em *Q-Learning*, e.g., **SR-DQN** e **DA-DDDQN**, tornam-se computacionalmente inviáveis em espaços de ação extensos, uma vez que requerem a

estimativa dos valores $Q(s, a)$ para todas as ações possíveis em cada estado. Para contornar essa limitação, o **NSE-A2C** adota uma abordagem baseada em políticas, combinando dois componentes distintos e complementares: o Ator, responsável por aprender diretamente a política estocástica $\pi(a|s)$, que define uma distribuição de probabilidade sobre as ações; e o Crítico, que estima o valor do estado $V(s)$, utilizado para orientar a atualização da política por meio de gradientes mais estáveis.

O diferencial do **NSE-A2C** em relação ao **A2C** tradicional reside na normalização dos estados e recompensas, e no uso de entropia como fator de regularização dinâmica. A normalização tem por objetivo reduzir a variância e estabilizar o treinamento ao longo do tempo, permitindo que o agente opere de forma mais consistente mesmo em ambientes de alta dimensionalidade. Isso é obtido ao ajustar cada vetor de estado com base em sua média e desvio padrão, suavizando flutuações abruptas entre observações consecutivas.

Além disso, o algoritmo incorpora um termo de entropia na função de perda do Ator, incentivando a exploração contínua de ações. Essa entropia é ponderada por um coeficiente dinâmico β , que decai progressivamente a cada episódio, assegurando que a política exploratória seja dominante nos estágios iniciais e gradualmente transite para uma política mais determinística conforme o agente converge para a solução ótima. A combinação dessas técnicas confere ao **NSE-A2C** maior robustez, estabilidade e eficiência durante o processo de aprendizado em ambientes complexos e com variabilidade elevada, como os simulados no **ns-3** com múltiplos **VANTs**.

O **NSE-A2C** busca maximizar a expectativa da recompensa acumulada utilizando a vantagem $A(s, a)$ como estimador da qualidade relativa de uma ação em relação ao valor esperado do estado. Essa vantagem é usada para ponderar a probabilidade da ação durante o ajuste da política, promovendo um aprendizado mais eficiente.

Durante o treinamento, a vantagem é estimada por $A_t = r + (1 - done) \cdot \gamma V(s') - V(s)$, que considera a recompensa imediata e o valor descontado do próximo estado. A normalização dos estados é realizada utilizando $\hat{s}_t = \frac{s_t - \mu}{\sigma + \nu}$, onde μ e σ representam a média e o desvio padrão das observações, respectivamente, e ν é um termo de suavização para evitar divisões por zero. Da mesma forma, as recompensas são normalizadas através de $\tilde{r} = \mathcal{R}_{norm}(r)$.

Para evitar a convergência prematura para políticas determinísticas, o algoritmo utiliza a perda de entropia, que estimula a exploração e a diversidade de experiências. O coeficiente de entropia β é ajustado de forma dinâmica ao longo dos episódios. O Crítico é otimizado minimizando o **MSE** entre a vantagem estimada e o valor atual do estado, utilizando a função de perda $\mathcal{L}_C = \frac{1}{T} \sum_t A_t^2$.

Algoritmo 5.3: *Normalized-State Entropy Advantage Actor-Critic (NSE-A2C)* para Alocação Dinâmica de Recursos em Redes IoT com VANT. [Criada pelo Autor.]

Entrada: Ambiente ns-3, \mathcal{S} , \mathcal{A} , γ , α , β_{\max} , β_{\min} , β_{decay} , \mathcal{B} , E_{\max} , T_{\max}

Saída: Política treinada π^* para alocação de VANTs

Inicialização:

Inicializar Redes *Ator* π_θ e *Crítico* V_ϕ com pesos θ e ϕ

Configurar otimizadores Adam para θ e ϕ

Inicializar normalizador de recompensas \mathcal{R}_{norm}

Configurar e inicializar ambiente ns-3 e obter s_0

para $e \leftarrow 1$ to E_{\max} faça

 para $t \leftarrow 1$ to T_{\max} faça

Política Estocástica:

 Normalizar estado: $\hat{s}_t \leftarrow \frac{s_t - \mu}{\sigma + \nu}$

 Obter política: $\pi(a_t | s_t) = \text{Softmax}(\pi_\theta(\hat{s}_t))$

 Selecionar ação $a_t \sim \pi(a_t | s_t)$

 Executar a_t e obter $(s', r, done)$

 Normalizar recompensa: $\tilde{r} = \mathcal{R}_{norm}(r)$

Cálculo da Vantagem:

 Determinar $A_t = \tilde{r} + (1 - done)\gamma V_\phi(s') - V_\phi(s_t)$

 Armazenar (s_t, a_t, r_t, A_t)

 se *done* então

 └ **break**

Atualização das Redes:

 Calcular perda do Crítico: $\mathcal{L}_C = \frac{1}{T} \sum_t A_t^2$

 Calcular perda do Ator: $\mathcal{L}_A = -\frac{1}{T} \sum_t \log \pi_\theta(a_t | s_t) A_t$

 Calcular perda de entropia: $\mathcal{L}_{ent} = -\beta \sum_t H(\pi_\theta(s_t))$

 Atualizar $\theta \leftarrow \theta - \alpha \nabla_\theta (\mathcal{L}_A + \mathcal{L}_{ent})$

 Atualizar $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_C$

Ajuste da Exploração:

 Atualizar coeficiente de entropia: $\beta \leftarrow \max(\beta_{\min}, \beta_{\max} \cdot \beta_{decay})$

 Reiniciar ambiente e obter estado inicial s_0

Retorne Política treinada π^*

Por outro lado, o Ator é otimizado maximizando a probabilidade das ações que resultaram em maiores vantagens, por meio da função de perda $\mathcal{L}_A = -\frac{1}{T} \sum_t \log \pi_\theta(a_t | s_t) A_t$, complementada pela perda de entropia $\mathcal{L}_{ent} = -\beta \sum_t H(\pi_\theta(s_t))$. As atualizações dos pesos das redes são realizadas utilizando o algoritmo de gradiente descendente: $\theta \leftarrow \theta - \alpha \nabla_\theta (\mathcal{L}_A + \mathcal{L}_{ent})$, e $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_C$, onde α representa a taxa

de aprendizado. Esse procedimento assegura que o Ator aprenda políticas mais eficientes e que o Crítico refine continuamente sua estimativa sobre o valor dos estados. O Algoritmo 5.3 detalha cada etapa do treinamento.

Apesar das melhorias introduzidas pelo NSE-A2C, como maior estabilidade e eficiência no aprendizado, o algoritmo ainda é sensível a grandes variações de atualização de políticas, o que pode comprometer a convergência em ambientes altamente dinâmicos. O algoritmo RG2E-PPO é então proposto com objetivo de superar essas limitações.

Reward-Guided and Generalized-Advantage Entropic PPO

O algoritmo RG2E-PPO representa uma evolução significativa em relação ao NSE-A2C, oferecendo maior estabilidade e eficiência no processo de aprendizado por reforço. Diferente de abordagens tradicionais, o RG2E-PPO foi desenvolvido para lidar com a sensibilidade nas atualizações de políticas, um desafio recorrente em algoritmos baseados em gradiente. No NSE-A2C, as atualizações da política são realizadas diretamente com base na vantagem estimada, o que, embora eficiente em muitos cenários, pode levar a mudanças abruptas e instáveis, especialmente em ambientes com alta complexidade e variação dinâmica.

Para superar essa limitação, o RG2E-PPO adota uma abordagem baseada em restrições, introduzindo um mecanismo de *clipping* na função de perda. Esse mecanismo atua limitando o grau de atualização da política a cada iteração, garantindo que as mudanças sejam graduais e seguras. Com isso, o algoritmo evita oscilações drásticas e melhora a estabilidade do aprendizado, promovendo uma convergência mais robusta e eficiente. Além disso, o RG2E-PPO equilibra a exploração e a exploração estratégica ao longo do treinamento, utilizando técnicas como o decaimento dinâmico do parâmetro de exploração e o cálculo de vantagens generalizadas (*Generalized Advantage Estimator* (GAE)). Essa combinação de estratégias torna o RG2E-PPO altamente adaptável e eficaz para problemas complexos. A função de perda do RG2E-PPO é definida na Equação 5-11,

Algoritmo 5.4: Reward-Guided and Generalized-Advantage Entropic Proximal Policy Optimization (RG2E-PPO) para Alocação Dinâmica de Recursos em Redes IoT com VANT

Entrada: Ambiente ns-3, \mathcal{S} , \mathcal{A} , γ , λ , α , β , \mathcal{B} , E_{max} , T_{max} , ϵ , ϵ_{min} , ϵ_{decay} , ϵ_{clip}

Saída: Política treinada π^* para alocação de VANTs

Inicialização:

Inicializar Redes de Política π_θ e de Valor V_ϕ com pesos θ e ϕ

Configurar otimizadores Adam para θ e ϕ , com regularização L2

Inicializar normalizador de recompensas \mathcal{R}_{norm}

Configurar e inicializar ambiente ns-3 e obter o s_0

para $e \leftarrow 1$ to E_{max} faça

 para $t \leftarrow 1$ to T_{max} faça

Seleção de Ação com Política Estocástica:

 Obter distribuição de ações $\pi(a_t|s_t)$ a partir da Rede de Política π_θ

 Selecionar ação $a_t \sim \pi(a_t|s_t)$ com probabilidade $(1 - \epsilon)$

 Com probabilidade ϵ , selecionar ação aleatória

 Atualizar $\epsilon \leftarrow \max(\epsilon_{min}, \epsilon \cdot \epsilon_{decay})$

 Executar ação a_t no ambiente e obter $(s', r, done)$

 Normalizar recompensa: $\tilde{r} = \mathcal{R}_{norm}(r)$

Cálculo da Vantagem com GAE:

 Determinar $A_t = \sum_{k=0}^{T-t} (\gamma\lambda)^k (\tilde{r}_{t+k} + \gamma V_\phi(s_{t+k+1}) - V_\phi(s_{t+k}))$

 Armazenar $(s_t, a_t, r_t, A_t, \pi(a_t|s_t))$ na memória

 se *done* então

 └ break

Atualização das Redes:

 Amostrar minibatch \mathcal{B} da memória

 para cada minibatch b em \mathcal{B} faça

 Calcular razão de probabilidade: $\rho_t = \frac{\pi_\theta(a_t|s_t)}{\pi_\theta^{old}(a_t|s_t)}$

 Calcular perda de política com clipping:

$\mathcal{L}_\pi = \frac{1}{T} \sum_t \min(\rho_t A_t, \text{clip}(\rho_t, 1 - \epsilon_{clip}, 1 + \epsilon_{clip}) A_t)$

 Calcular perda de valor: $\mathcal{L}_V = \frac{1}{T} \sum_t (V_\phi(s_t) - R_t)^2$

 Calcular perda de entropia: $\mathcal{L}_{ent} = -\beta \sum_t H(\pi_\theta(s_t))$

 Atualizar $\theta \leftarrow \theta - \alpha \nabla_\theta (\mathcal{L}_\pi + \mathcal{L}_{ent})$

 Atualizar $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_V$

 Reiniciar ambiente e obter estado inicial s_0

Retorne Política treinada π^*

$$\mathcal{L}_{PPO}(\theta) = \mathbb{E}_t \left[\min(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1 - \epsilon_{clip}, 1 + \epsilon_{clip})A_t) \right], \quad (5-11)$$

onde, $r_t(\theta)$ é a razão entre a nova e a antiga política, calculada como $\frac{\pi_\theta(a_t|s_t)}{\pi_\theta^{old}(a_t|s_t)}$, A_t é a vantagem estimada no tempo t , e a função *clip* restringe a atualização da razão de probabilidade para o intervalo $[1 - \epsilon_{clip}, 1 + \epsilon_{clip}]$, garantindo que as atualizações sejam graduais e estáveis. Além do *clipping*, o algoritmo utiliza a **GAE** para reduzir a variância das estimativas de vantagem e uma perda de entropia para incentivar a exploração contínua durante o treinamento. O coeficiente de entropia β é ajustado dinamicamente ao longo dos episódios, e o parâmetro de exploração ϵ decai progressivamente, equilibrando a exploração e a exploração estratégica.

O Algoritmo 5.4 apresenta a implementação detalhada do **RG2E-PPO**. O algoritmo inicia com a configuração das redes de política e valor, juntamente com seus respectivos otimizadores com regularização $L2$. Durante cada episódio, o agente interage com o ambiente **ns-3**, selecionando ações com base em uma política estocástica $\pi(a_t|s_t)$. A seleção de ações é balanceada por uma estratégia ϵ -greedy, onde a escolha aleatória de ações reduz gradualmente ao longo dos episódios, controlada por ϵ_{decay} . As recompensas são normalizadas para estabilizar o aprendizado, e as vantagens são calculadas utilizando o método de **GAE**. Após o término de cada episódio, as redes de política e valor são atualizadas em *minibatches*, utilizando a perda de política com *clipping*, a perda de valor para refinar as estimativas de recompensa e a perda de entropia para promover a diversidade de ações. Esse ciclo é repetido ao longo de todos os episódios, assegurando uma aprendizagem robusta e eficiente.

A formulação apresentada neste capítulo estabelece uma base sólida para a aplicação de aprendizado por reforço em cenários de alocação dinâmica de recursos com **VANTs**. Os algoritmos desenvolvidos, e.g., **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**, foram concebidos com foco em estabilidade, eficiência e robustez durante o processo de aprendizado. Cada abordagem foi projetada para lidar com desafios específicos do domínio, como a superestimação dos valores Q no **SR-DQN** e a redução da variância nas atualizações por meio da estimativa de vantagem generalizada no **NSE-A2C**. A Seção 5.3, descreve em detalhes as estratégias de avaliação adotadas, visando validar o desempenho das políticas aprendidas no escopo desse trabalho.

5.3 Avaliações

Nesta seção, são apresentadas as estratégias adotadas para a análise de desempenho das soluções desenvolvidas. Diferentemente da formulação exata baseada em **MILP**, descrita no Capítulo 4, as soluções aqui avaliadas foram obtidas a partir do treinamento de agentes **DRL**, e.g., **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**. Esses modelos têm por objetivo aprender estratégias ótimas de alocação de recursos por meio da inte-

ração com um ambiente simulado, utilizando o [ns-3](#) para capturar as dinâmicas reais da comunicação sem fio.

O treinamento dos agentes foi conduzido com o propósito de otimizar múltiplos critérios, incluindo a maximização da [QoS](#), a minimização do número de [VANTs](#) necessários e a adaptação eficiente às variações na distribuição espacial dos dispositivos [IoT](#). O processo de aprendizado possibilitou o desenvolvimento de políticas adaptativas, ajustadas continuamente às mudanças observadas no ambiente.

Os experimentos realizados foram planejados para representar um cenário urbano realista, levando em consideração as regulamentações da aviação civil e as características específicas das redes [LoRaWAN](#) e [5G](#). O objetivo principal consistiu em avaliar a capacidade dos algoritmos de [DRL](#) de aprender e aplicar estratégias eficientes de alocação dinâmica de recursos, garantindo desempenho competitivo em relação às soluções exatas, com ganhos em escalabilidade e adaptabilidade.

Para avaliar a eficácia das soluções propostas, foram consideradas métricas como a taxa de entrega de pacotes, a vazão e o atraso médios das comunicações, o tempo de convergência do treinamento e a eficiência do modelo em diversos cenários. Também foram investigadas a estabilidade das políticas aprendidas e sua capacidade de generalização para diferentes topologias de rede. Adicionalmente, são discutidas as limitações dos métodos e os contextos mais apropriados para sua aplicação.

Metodologia de Treinamento e Avaliação

O processo de treinamento e avaliação das soluções baseadas em aprendizado por reforço, utilizando os agentes de [DRL](#) que implementam os algoritmos [SR-DQN](#), [DA-DDDQN](#), [NSE-A2C](#) e [RG2E-PPO](#), foram concebidos para assegurar a robustez e a capacidade de generalização das políticas aprendidas. A Figura 5.3 ilustra a metodologia desenvolvida, que integra o agente de aprendizado por reforço ao ambiente de simulação [ns-3](#) por meio do *middleware ns3Gym* [[Gawłowicz e Zubow 2019](#)], que atua como uma *Application Programming Interface (API)* para a integração do agente com o simulador [ns-3](#), utilizando comunicação assíncrona por meio de *sockets ZeroMQ Messaging Library (ZMQ)* e implementado sobre a biblioteca de serialização *Protocol Buffers* [[Google 2025](#)]. O *ns3Gym* permite a troca contínua de informações e decisões ao longo das iterações de treinamento e avaliação.

Treinamento

As seções a seguir detalham as etapas envolvidas no processo de treinamento dos agentes de aprendizado por reforço, conforme representado na Figura 5.3. Cada componente do fluxo, desde a interação entre agente e ambiente até a atualização das

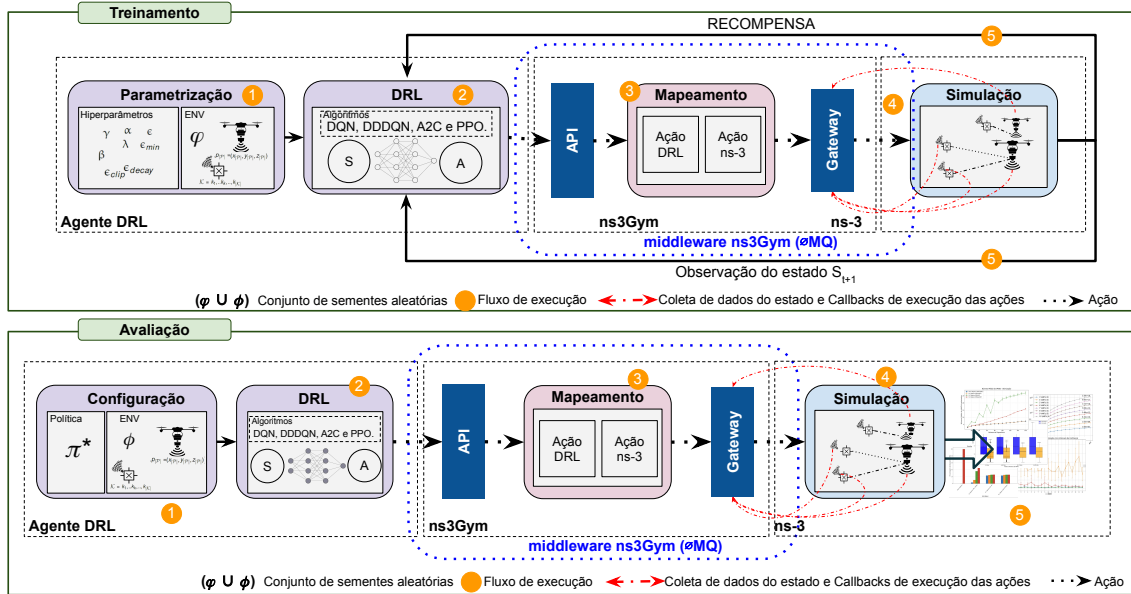


Figura 5.3: Metodologia do processo de treinamento e avaliação das soluções baseadas em aprendizado de máquina. [Criada pelo Autor.]

políticas, é descrito, com destaque para os mecanismos responsáveis pelo aprendizado e pela adaptação das decisões ao longo dos episódios.

Parametrização

O processo de treinamento do agente tem início com a etapa de parametrização, responsável pela configuração detalhada do ambiente de simulação, a qual abrange tanto os hiperparâmetros do agente quanto os parâmetros operacionais do simulador. O agente de DRL em treinamento, e.g., SR-DQN, DA-DDDQN, NSE-A2C ou RG2E-PPO, é inicializado com um conjunto de hiperparâmetros previamente definidos, tais como o fator de desconto (γ), o fator de suavização para o GAE (λ), os coeficientes de exploração (ϵ) e de entropia (β), entre outros.

Na sequência, são configurados os conjuntos de dispositivos, suas posições iniciais e as posições candidatas para a implantação dos VANTs. Os LoRa-EDs, representados pelo conjunto \mathcal{K} , são distribuídos em uma área urbana externa de 20 km^2 , utilizando um padrão log-normal que modela de forma realista a densidade espacial do tráfego em cenários urbanos. As posições candidatas à implantação dos VANTs, representadas pelo conjunto \mathcal{P} , são organizadas em uma malha tridimensional regular, com pontos uniformemente espaçados e fixados a uma altitude de 45 metros, em conformidade com as diretrizes da ANAC. Essas configurações seguem as mesmas diretrizes adotadas nos experimentos com soluções exatas, conforme descrito na Seção 4.4 do Capítulo 4.

Todas as configurações, tanto dos hiperparâmetros quanto do ambiente do si-

mulador, constituem a base de entrada para o ciclo de treinamento dos agentes de **DRL**, orientando o processo de aprendizado e a adaptação das políticas às condições dinâmicas da rede.

Configuração do agente de DRL

Na sequência, a etapa denominada *DRL* é responsável pelo controle de cada agente em treinamento. Nessa etapa, são definidos os estados \mathcal{S} , as ações \mathcal{A} e configurada a estrutura da **NN** do algoritmo em execução, e.g., **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**. Cada configuração foi cuidadosamente especificada para garantir a eficiência e a estabilidade do processo de aprendizado, considerando os desafios específicos do problema. A seguir, são apresentadas as configurações adotadas para o treinamento de cada algoritmo.

Configurações de treinamento do algoritmo SR-DQN: O treinamento do agente **SR-DQN** utilizou uma **NN** para a aproximação da função $Q(s, a)$, cuja atualização segue a minimização da função de perda

$$\mathcal{L}(\theta) = \mathbb{E}_{(s,a,r,s')} [(y - Q(s, a; \theta))^2], \quad (5-12)$$

com o valor de referência y definido por

$$y = r + \gamma \max_{a'} Q(s', a'; \theta'). \quad (5-13)$$

A **NN** do agente **SR-DQN** foi configurada como um modelo sequencial composto por três camadas densas. A camada de entrada recebe um vetor de dimensão correspondente ao espaço de estados, seguida por duas camadas ocultas com 256 neurônios cada, ativadas por funções *Rectified Linear Unit* (**ReLU**), a fim de introduzir não linearidade ao aprendizado. A camada de saída possui um número de neurônios igual ao total de ações possíveis, fornecendo as estimativas dos valores Q para cada ação no estado atual.

Adotou-se o uso de um *Replay Buffer* no agente **SR-DQN** para armazenar e reutilizar experiências passadas, promovendo maior eficiência no aprendizado ao reduzir a correlação entre as amostras de treinamento. Implementado com *Prioritized Experience Replay* (**PER**), o *buffer* atribui prioridades às experiências com base no erro de aprendizado, ajustado pelo parâmetro α , permitindo que transições mais relevantes tenham maior probabilidade de serem amostradas. Durante o treinamento, um lote de experiências é amostrado com pesos de importância ajustados por β , garantindo um balanceamento entre exploração e estabilidade do aprendizado. Além disso, o tamanho do *buffer* é definido proporcionalmente à complexidade do espaço de estados, com um limite má-

ximo de 100.000 transições, assegurando um armazenamento eficiente sem comprometer a escalabilidade do modelo.

Os hiperparâmetros utilizados estão resumidos na Tabela 5.2.

Tabela 5.2: Hiperparâmetros utilizados no treinamento do SR-DQN. [Criada pelo Autor.]

Parâmetro	Valor
Fator de desconto (γ)	0,999
Taxa de aprendizado	2×10^{-4}
ϵ inicial	1,0
ϵ mínimo	0,5
Decaimento de ϵ	0,9975
α	0,6
β	0,4
Tamanho do <i>replay buffer</i>	$\min(10 * \mathcal{S} , 100.000)$
Função de ativação	ReLU
Camada de entrada da NN	$ \mathcal{S} $
Camadas ocultas da NN	3 camadas - 256 neurônios
Camada de saída da NN	$ \mathcal{A} $ neurônios
Frequência de atualização da rede-alvo	a cada 5 episódios

Configurações de treinamento do algoritmo DA-DDDQN: O DA-DDDQN combina duas melhorias fundamentais em relação ao DDQN tradicional: a arquitetura *Dueling Network* e a técnica *Double Q-Learning*. A abordagem *Dueling* separa a estimativa do valor do estado $V(s)$ da vantagem das ações $A(s, a)$, permitindo que a rede neural diferencie estados relevantes daqueles irrelevantes para a tomada de decisão. A estimativa final dos valores $Q(s, a)$ é obtida por meio de

$$Q(s, a) = V(s) + \left(A(s, a) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s, a') \right), \quad (5-14)$$

onde média das vantagens é subtraída para evitar que a função de valor $V(s)$ interfira indevidamente na diferenciação entre ações. Por sua vez, a técnica *Double Q-Learning* reduz a superestimação dos valores Q ao utilizar duas redes distintas, a rede principal seleciona a melhor ação no próximo estado, enquanto a rede-alvo avalia o valor dessa ação. A atualização dos valores segue a equação

$$y = r + \gamma Q(s', \arg \max_{a'} Q(s', a'; \theta); \theta'), \quad (5-15)$$

em que a escolha da ação é realizada com os pesos da rede principal θ , enquanto a estimativa do valor da ação é computada pela rede-alvo θ' . Essa abordagem melhora

a estabilidade do aprendizado e reduz o viés na seleção das ações, tornando a política aprendida mais robusta e eficiente.

A rede neural utilizada adota a arquitetura *Dueling Network*, que separa explicitamente a estimativa do valor do estado da vantagem das ações, permitindo uma diferenciação mais eficiente entre estados relevantes e irrelevantes. Sua estrutura inicia-se com uma camada de entrada, cuja dimensão corresponde ao espaço de estados do problema. Essa entrada é processada por duas camadas totalmente conectadas, cada uma com 256 neurônios e ativação **ReLU**, responsáveis por extrair representações mais abstratas do estado.

Tabela 5.3: *Hiperparâmetros utilizados no treinamento do DA-DDDQN. [Criada pelo Autor.]*

Parâmetro	Valor
Fator de desconto (γ)	0,99
Taxa de aprendizado	1×10^{-3}
ϵ inicial	1,0
ϵ mínimo	0,01
Decaimento de ϵ	0,995
Tamanho do <i>Replay Buffer</i>	100.000 transições
Algoritmo de otimização	Adam
Função de perda	MSELoss
Função de ativação	ReLU
Camada de entrada	$ \mathcal{S} $
Camadas ocultas	2 com 256 neurônios
Camada intermediária de Valor	128 neurônios
Camada de saída de Valor	1 neurônio para $V(s)$
Camada intermediária de Vantagem	128 neurônios
Camada de saída de Vantagem	$ \mathcal{A} $ neurônios
Frequência de atualização da rede-alvo	Após cada episódio de treino

Após essas camadas compartilhadas, a rede é dividida em dois fluxos distintos, o fluxo de valor, que estima o valor do estado $V(s)$, e o fluxo de vantagem, que calcula a vantagem associada a cada ação $A(s, a)$. No fluxo de valor, as informações são processadas por uma camada intermediária de 128 neurônios, seguida de uma camada de saída com um único neurônio, representando a estimativa escalar de $V(s)$. Já no fluxo de vantagem, a rede passa por uma camada intermediária de 128 neurônios, seguida de uma camada de saída com $|\mathcal{A}|$ neurônios, correspondente ao número de ações possíveis, fornecendo a estimativa de $A(s, a)$. Os hiperparâmetros utilizados no **DA-DDDQN** estão descritos na Tabela 5.3.

Configurações de treinamento do algoritmo NSE-A2C: O algoritmo **NSE-A2C** utiliza duas redes neurais principais, o Ator e o Crítico. Essa abordagem permite a otimização direta da política do agente, utilizando a estimativa da vantagem para reduzir a

variância nas atualizações e acelerar o aprendizado. A vantagem é calculada a partir da recompensa imediata r , do valor do próximo estado $V(s')$ e do estado atual $V(s)$, conforme

$$A(s, a) = r + \gamma V(s') - V(s). \quad (5-16)$$

A função de perda total do modelo é composta por três termos

$$\mathcal{L} = \mathcal{L}_A + c_1 \mathcal{L}_C - c_2 \mathcal{L}_{ent}, \quad (5-17)$$

onde $\mathcal{L}_A = -\mathbb{E}_t [\log \pi(a_t | s_t) A_t]$ representa a perda do ator, que ajusta a política para maximizar a probabilidade das ações mais vantajosas, $\mathcal{L}_C = \frac{1}{2} (R_t - V(s_t))^2$ corresponde à perda do crítico, que minimiza o erro quadrático da estimativa de valor, e $\mathcal{L}_{ent} = -\beta H(\pi(s_t))$ é a perda de entropia, que incentiva a diversidade na escolha das ações para evitar a convergência prematura da política.

A arquitetura da rede neural do Ator é composta por três camadas densas. A entrada, de dimensão correspondente ao espaço de estados, é processada por duas camadas ocultas totalmente conectadas, cada uma com 256 neurônios e ativação *Tanh*. A saída contém $|\mathcal{A}|$ neurônios, onde $|\mathcal{A}|$ representa o número de ações possíveis, e aplica a ativação *Softmax*, garantindo que a saída represente uma distribuição de probabilidades

$$\pi(a|s) = \text{Softmax}(W_3 \cdot \text{Tanh}(W_2 \cdot \text{Tanh}(W_1 \cdot s + b_1) + b_2) + b_3). \quad (5-18)$$

A rede do Crítico, utilizada para estimar $V(s)$, segue uma estrutura semelhante, com duas camadas ocultas de 256 neurônios ativadas por *ReLU*. A camada final contém um único neurônio, fornecendo uma estimativa escalar do valor do estado

$$V(s) = W_3 \cdot \text{ReLU}(W_2 \cdot \text{ReLU}(W_1 \cdot s + b_1) + b_2) + b_3. \quad (5-19)$$

Nas redes neurais do algoritmo *NSE-A2C*, funções de ativação distintas são empregadas no Ator e no Crítico, de acordo com os objetivos e características de cada componente. A rede do Ator adota a função de ativação *Tanh* nas camadas ocultas, que fornece saídas no intervalo $(-1, 1)$. Essa escolha favorece a suavidade da política estocástica aprendida, ao evitar saturações abruptas e permitir que a distribuição de probabilidades sobre as ações varie de forma mais controlada, especialmente quando combinada com uma camada de saída com ativação *Softmax*. A suavidade promovida pelo *Tanh* contribui para uma política mais exploratória nas fases iniciais do treinamento e mais refinada nas etapas finais, à medida que a entropia é reduzida.

Por outro lado, a rede do Crítico utiliza a função de ativação *ReLU* nas camadas ocultas. O *ReLU* é amplamente adotado em tarefas de regressão devido à sua simplicidade computacional e à capacidade de preservar gradientes positivos, o que favorece a

propagação eficiente de informações durante o treinamento. Como o Crítico busca estimar o valor escalar de um estado $V(s)$, a **ReLU** contribui para uma resposta mais estável e rápida a mudanças nos dados de entrada, evitando a saturação que poderia comprometer o aprendizado da função de valor. A camada de saída do Crítico, por sua vez, é linear, retornando uma estimativa escalar contínua do valor do estado.

Essa separação no uso das funções de ativação reflete a natureza distinta das tarefas desempenhadas por cada rede: enquanto o Ator modela uma política probabilística, o Crítico realiza uma estimativa determinística do valor esperado, exigindo respostas mais amplas e diretas do modelo.

Para favorecer um aprendizado estável, os pesos das redes são inicializados pelo método *Kaiming Uniform* [He et al. 2015], em conjunto com a função de ativação **ReLU**, o que melhora a propagação dos gradientes e minimiza problemas de saturação.

Durante o treinamento do agente **NSE-A2C**, foram empregadas estratégias para estabilizar o aprendizado e melhorar a eficiência da política aprendida. A perda de entropia foi incorporada ao modelo como termo regularizador, incentivando a diversidade na seleção de ações e prevenindo a convergência prematura para políticas determinísticas. O coeficiente de entropia β foi ajustado dinamicamente, iniciando com um valor elevado para promover exploração e reduzido progressivamente ao longo do treinamento. Além disso, a normalização da recompensa foi aplicada utilizando uma média móvel exponencial com fator de suavização $\alpha = 0,99$, garantindo que as recompensas fossem escaladas de maneira controlada, reduzindo a variância dos gradientes e melhorando a estabilidade do aprendizado. Para evitar explosões de gradiente e assegurar uma atualização mais controlada dos pesos da rede neural, foi empregada a regularização do gradiente por meio da técnica de *clipping*, limitando o valor máximo do gradiente a 1,0 ($max_norm = 1,0$). Os principais hiperparâmetros utilizados no treinamento do **NSE-A2C** estão detalhados na Tabela 5.4.

Configurações de treinamento do algoritmo RG2E-PPO: O algoritmo **RG2E-PPO** foi configurado com o objetivo de promover atualizações estáveis e eficientes da política, utilizando a técnica de *clipping* para restringir alterações abruptas nos parâmetros. Essa abordagem controla o impacto das atualizações, assegurando uma progressão mais suave no aprendizado e maior robustez contra oscilações indesejadas.

Para formalizar essa estratégia, a função de perda central do **RG2E-PPO** é definida por,

$$\mathcal{L}_{PPO} = \mathbb{E}_t \left[\min \left(r_t(\theta) A_t, \text{clip} \left(r_t(\theta), 1 - \epsilon_{clip}, 1 + \epsilon_{clip} \right) A_t \right) \right], \quad (5-20)$$

onde, $r_t(\theta)$ representa a razão entre a nova e a antiga política, e é calculada como $r_t(\theta) = \frac{\pi_{\theta}(a_t|s_t)}{\pi_{\theta^{old}}(a_t|s_t)}$, e A_t é a vantagem estimada. O parâmetro ϵ_{clip} define um intervalo de

Tabela 5.4: Hiperparâmetros utilizados no treinamento do NSE-A2C. [Criada pelo Autor.]

Parâmetro	Valor
Fator de desconto (γ)	0,90
Taxa de aprendizado do ator	$1,5 \times 10^{-5}$
Taxa de aprendizado do crítico	2×10^{-5}
Algoritmo de otimização	Adam
Decaimento de peso ($L2$)	1×10^{-5}
Função de ativação do ator	Tanh
Função de ativação do crítico	ReLU
Estrutura da rede do ator	Entrada $ S $, 2 com 256 neurônios, e saída $ A $
Estrutura da rede do crítico	Entrada $ S $, 2 com 256 neurônios, e saída a_{t+1}
Inicialização de pesos	<i>Kaiming Uniform</i>
Coefficiente de entropia inicial (β)	0,035
Coefficiente de entropia final	0,01
Taxa de decaimento da entropia	0,99
Normalização de recompensa	Média móvel exponencial ($\alpha = 0,99$)
ϵ inicial	1×10^{-8}
ϵ mínimo	0,01
Decaimento de ϵ	0,995
Regularização de gradiente	<i>Clipping</i> com $\max_norm = 1,0$

confiança para a atualização da política, limitando $r_t(\theta)$ e, conseqüentemente, impedindo mudanças excessivas entre as iterações.

Para estimar A_t , foi empregada a **GAE**, que equilibra o viés e a variância na estimativa da função de vantagem por meio dos hiperparâmetros de fator de desconto (γ) e fator de suavização temporal (λ),

$$A_t = \sum_{k=0}^{T-t} (\gamma\lambda)^k [r_{t+k} + \gamma V(s_{t+k+1}) - V(s_{t+k})]. \quad (5-21)$$

Assim, a função de perda total combina os três componentes,

$$\mathcal{L}_{total} = \mathcal{L}_{PPO} + c_1 \mathcal{L}_V + c_2 \mathcal{L}_{ent}, \quad (5-22)$$

onde \mathcal{L}_V representa a perda do Crítico, minimizando o erro quadrático da estimativa de valor $V(s)$, e \mathcal{L}_{ent} corresponde à perda de entropia, que estimula a diversidade de ações e previne a convergência precoce para políticas determinísticas.

O algoritmo de **RG2E-PPO** utiliza duas redes neurais separadas, o Ator, responsável por aprender a política $\pi(a|s)$, e o Crítico, encarregado de estimar o valor do estado $V(s)$. Ambas compartilham uma arquitetura composta por duas camadas ocultas densas com 128 neurônios cada, ativadas por funções **ReLU**. A rede do Ator possui uma camada de saída com ativação *Softmax*, que transforma as saídas em uma distribuição de probabi-

lidades sobre o espaço de ações. Já o Crítico termina em uma camada densa com ativação linear, retornando uma estimativa escalar contínua de valor.

Para melhorar a propagação dos gradientes e manter a estabilidade do treinamento, os pesos das redes são inicializados com o método *Kaiming Normal* [He et al. 2015]. Esta técnica assume uma distribuição normal dos pesos, com média zero e variância adaptada à quantidade de neurônios da camada anterior. Ela é especialmente adequada para redes com ativação **ReLU**, pois preserva a variância dos sinais de entrada ao longo das camadas, reduzindo o risco de explosão ou desaparecimento do gradiente. Em contraste, o *Kaiming Uniform* utiliza uma distribuição uniforme para a inicialização, sendo uma alternativa com características similares, mas que distribui os pesos em torno de um intervalo simétrico.

Além disso, foi empregada a regularização *L2* (*weight decay*) nos otimizadores das redes para evitar o sobreajuste. Esse mecanismo adiciona um termo penalizador proporcional à norma *L2* dos pesos na função de perda, limitando o crescimento dos parâmetros da rede. Como resultado, a regularização *L2* contribui para uma generalização mais eficaz da política, especialmente em ambientes com alta variabilidade estrutural e ruído. Os hiperparâmetros utilizados no treinamento do **RG2E-PPO** estão detalhados na Tabela 5.5.

Tabela 5.5: Hiperparâmetros utilizados no treinamento do RG2E-PPO. [Criada pelo Autor.]

Parâmetro	Valor
Fator de desconto (γ)	0,99
λ (GAE)	0,95
Taxa de aprendizado do ator	3×10^{-4}
Taxa de aprendizado do crítico	2×10^{-4}
Estrutura da rede de política	Entrada $ S $, 2 de 128 neurônios, e saída $ A $
Estrutura da rede de valor	Entrada $ S $, 2 de 128 neurônios, e saída a_{t+1}
Inicialização de pesos	<i>Kaiming Uniform</i>
ϵ_{clip}	0,2
Coefficiente de entropia inicial	0,1
Coefficiente de entropia final	0,01
Decaimento da entropia	0,995
ϵ inicial	1,0
ϵ mínimo	0,5
Decaimento de ϵ	0,999
Algoritmo de otimização	Adam com regularização <i>L2</i>
Função de ativação	ReLU
Tamanho do <i>mini-batch</i>	128
Clipping de gradiente	Max_norm =1,0

Mapeamento de Ações do Agente para Ações do Simulador

O treinamento do agente de **DRL** requer interação contínua entre a política aprendida e o ambiente de simulação. Para isso, foi desenvolvido um mecanismo de mapeamento entre as ações tomadas pelo agente e as ações executadas no simulador **ns-3** por meio do *ns3Gym*.

A cada iteração, o agente recebe uma observação do estado do ambiente S_t e, com base na política $\pi(a|s)$, realiza a seleção de uma ação. O *middleware* traduz essa ação em comandos compatíveis com o simulador **ns-3**, assegurando que as decisões do agente sejam aplicadas ao modelo de rede sem fio simulado. Após a execução da ação, o simulador fornece um novo estado S_{t+1} e uma recompensa \mathcal{R}_t , que são utilizados na atualização da política do agente.

A integração por meio do *ns3Gym* padroniza a comunicação entre o agente de **DRL** e o ambiente simulado no **ns-3**. O ambiente customizado implementa métodos fundamentais, como `reset()`, que inicializa a simulação e retorna o estado inicial, e `step(action)`, responsável por aplicar a ação recebida, atualizar o ambiente e retornar a nova observação, a recompensa e outras informações auxiliares. Essa arquitetura modular permite a substituição ou reconfiguração da política do agente sem necessidade de alterações na lógica interna do simulador, promovendo maior flexibilidade e reutilização do ambiente em diferentes experimentos.

Simulação e Retorno de Recompensas ao Agente

A etapa de simulação representa o núcleo da interação entre o agente de aprendizado por reforço e o ambiente de rede simulado. Após a seleção de uma ação, inicia-se um fluxo que envolve a tradução, execução e resposta da simulação, o qual retroalimenta o processo de aprendizado.

Uma ação é encaminhada para execução no simulador por meio da **API** de integração. A etapa de mapeamento recebe essa ação e a traduz para um comando compatível com o **ns-3**. Para garantir a correta aplicação, o mapeamento atua como intermediário na camada de execução do **ns-3**, convertendo as ações definidas pelo agente em parâmetros do simulador, e.g., transformar a ação escalar gerada pelo agente de **DRL** em uma ação de movimentação dos **VANTs**, assegurando que a decisão seja interpretada corretamente pelo simulador.

O *gateway* do ambiente é responsável por encaminhar a ação traduzida para execução no simulador. Uma vez recebida e mapeada, a ação é enviada ao ambiente de simulação **ns-3**, que, previamente configurado, executa por um intervalo de tempo definido, durante o qual processa os efeitos da ação no cenário de rede. Isso inclui a atualização do posicionamento dos **VANTs**, o cálculo das métricas de **QoS**, e.g., latência,

vazão e taxa de entrega de pacotes, além da verificação de eventos indesejados, como colisões ou saídas de área.

Durante essa fase, o **ns-3** também coleta informações sobre o novo estado do ambiente s_{t+1} , o qual é posteriormente retornado como parte da observação da transição de estado. Ao final da execução da simulação, é realizado o cálculo da recompensa r_t , com base no desempenho da ação executada. A função de recompensa considera critérios como melhoria ou degradação da **QoS**, penalidades por colisões entre **VANTs** ou por saída da área de cobertura, e bonificações por decisões que favoreçam conectividade eficiente e distribuição equilibrada de recursos.

Avaliação

Após o treinamento dos agentes de **DRL**, a etapa de avaliação é realizada com o objetivo de validar a robustez e a capacidade de generalização das políticas aprendidas. Essa fase é essencial para a análise do desempenho dos agentes em cenários distintos daqueles utilizados durante o treinamento, utilizando políticas fixas e sem exploração estocástica. A avaliação é conduzida a partir de um conjunto de sementes aleatórias ($\varphi \cup \phi$), em que ϕ representa as sementes consideradas para a avaliação, distintas daquelas empregadas no treinamento (φ), assegurando a diversidade dos cenários testados. O fluxo de avaliação segue uma estrutura modular composta pela configuração do ambiente, inicialização da política aprendida (π^*), mapeamento de ações, execução da simulação e coleta de métricas de desempenho, conforme ilustrado na Figura 5.3. A seguir, cada etapa é detalhada.

Configuração da Avaliação

A avaliação é conduzida com base nas soluções iniciais geradas pelos modelos de otimização baseados em **MILP**, as quais fornecem a estrutura de referência para a configuração dos cenários simulados no ambiente **ns-3**. A partir dessas configurações, o agente de **DRL** executa sua política aprendida π^* em episódios independentes, sem qualquer atualização dos parâmetros da rede. O processo de avaliação caracteriza-se, portanto, por operar de forma determinística, com a exploração desativada e o uso exclusivo da política final treinada.

Para cada conjunto de passos em um episódio, os **VANTs** são movimentados com base nas ações definidas pela política π^* . A cada episódio, um novo cenário é inicializado por meio de sementes aleatórias ϕ , distintas daquelas utilizadas no treinamento (φ), assegurando a diversidade dos ambientes avaliados e a verificação da capacidade de generalização da política.

Durante a execução, o agente interage com o ambiente simulado, recebe observações do estado atual s_t , seleciona uma ação $a_t = \pi^*(s_t)$, aplica essa ação no ambiente por meio do *middleware* e obtém uma nova observação s_{t+1} e a recompensa r_t . Como não há aprendizado nesta fase, os dados coletados são registrados apenas para análise posterior.

A configuração de cada episódio contempla a parametrização da rede, incluindo os dispositivos **LoRa-ED**, as posições candidatas para os **VANTs** e os parâmetros de comunicação. Essa abordagem garante variações topológicas, dinâmicas de interferência e distribuição espacial dos dispositivos, resultando em uma avaliação mais robusta, imparcial e representativa de cenários reais.

Configuração do Agente de DRL

Durante a avaliação, o agente executa exclusivamente a política aprendida π^* , sem realizar qualquer atualização nos pesos da rede neural. Diferentemente da fase de treinamento, a exploração é completamente desativada, e as ações são selecionadas de forma determinística, com base nas estimativas fornecidas pela rede.

A arquitetura da rede, os parâmetros finais do modelo treinado e a política π^* são previamente carregados, assegurando que a execução reproduza fielmente o comportamento aprendido. Esse procedimento garante que a avaliação reflita apenas a capacidade de generalização da política, sem interferência de adaptações ou ajustes durante os episódios.

Mapeamento das Ações

Após a seleção de uma ação pela política π^* , o processo de mapeamento é realizado por meio do *middleware* integrado ao **ns-3**. A ação a_t é encaminhada ao *proxy* de mapeamento, que converte a decisão oriunda do espaço de ação do agente em comandos interpretáveis pelo simulador, os quais são aplicados ao reposicionamento dos **VANTs**. Esse mapeamento garante que a ação, expressa em alto nível, seja corretamente interpretada e aplicada no ambiente simulado, respeitando a semântica da política aprendida.

Simulação

Com a ação aplicada ao ambiente, o simulador **ns-3** executa a próxima etapa da simulação. Essa execução reflete os efeitos da decisão tomada pela política π^* , com atualização do estado da rede, incluindo métricas como atraso, taxa de entrega de pacotes, quantidade e posicionamento dos **VANTs** e **QoS**.

Durante a simulação, as informações sobre o novo estado s_{t+1} são coletadas pelo *middleware*, e a recompensa r_t associada à ação executada é calculada com base na mesma função de recompensa utilizada durante o treinamento. Como não há aprendizado

nesta fase, os dados obtidos são armazenados exclusivamente para análise posterior, sem qualquer atualização da política.

Coleta e Análise dos Resultados

Ao final de cada episódio de avaliação, as principais métricas de desempenho obtidas com a política π^* são registradas. Esses resultados são representados por gráficos que ilustram a eficácia da política aprendida em diferentes cenários, e.g., PDR, QoS, atraso, vazão e número de VANTs utilizados.

A estrutura metodológica descrita nesta seção permitiu a execução de um processo de avaliação sistemático, baseado na política previamente treinada e executada em cenários diversos e realistas. A seguir, são apresentados na Seção 5.4 os resultados obtidos, com o objetivo de analisar o desempenho dos agentes de DRL nas diferentes configurações experimentais. Essa análise contempla múltiplas métricas de desempenho, permitindo comparar a eficácia das abordagens propostas e identificar padrões relevantes quanto à generalização, robustez e eficiência das soluções em ambientes dinâmicos.

5.4 Resultados

Esta seção apresenta os resultados obtidos com os algoritmos de DRL durante a fase de treinamento e na etapa de avaliação. Os dados obtidos são analisados sob diversas métricas de desempenho, com o objetivo de validar a eficácia, estabilidade e capacidade de generalização das políticas aprendidas em ambientes simulados com VANTs para redes IoT.

Comportamento dos Algoritmos no Treinamento

Esta subseção apresenta uma análise do desempenho dos algoritmos SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO ao longo da fase de treinamento. São avaliadas métricas como a evolução da recompensa média por episódio, o tempo de convergência e a estabilidade das políticas aprendidas, permitindo comparar a eficiência e a consistência de cada abordagem durante o processo de aprendizado.

Análise de Desempenho do Treinamento do Algoritmo SR-DQN

Nesta seção, são apresentados os resultados obtidos com a aplicação do algoritmo SR-DQN ao problema de posicionamento de VANTs atuando como gateways em um ambiente de simulação ns-3. O ambiente simulado foi configurado com uma malha de 11×11 posições candidatas (121 no total) e 50 LoRa-EDs. Foram avaliados quatro cenários distintos quanto à quantidade de VANTs (2, 3, 4 e 5), sendo executados 300 episódios

de treinamento por configuração. A Figura 5.4 apresenta os gráficos correspondentes aos indicadores de tempo de execução por episódio, perda média acumulada e recompensa média acumulada.

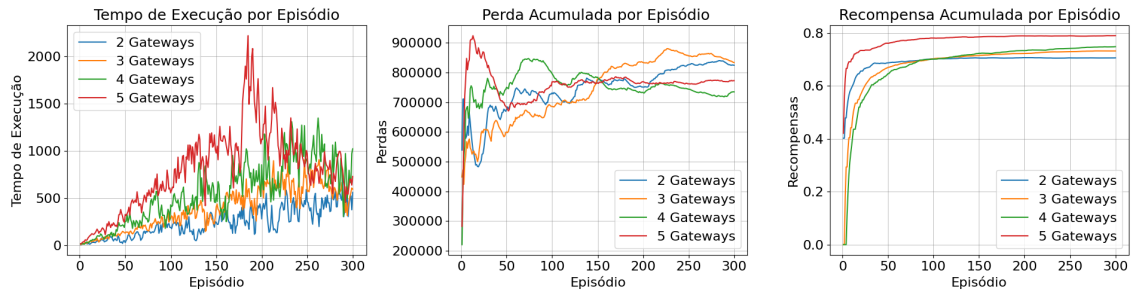


Figura 5.4: Desempenho do SR-DQN para 2, 3, 4 e 5 VANTs.
[Criada pelo Autor.]

Tempo de Execução por Episódio: O gráfico à esquerda da Figura 5.4 indica que o tempo de execução por episódio aumenta de forma consistente com o número de VANTs. A configuração com 2 VANTs apresenta os menores tempos médios, permanecendo majoritariamente abaixo de 500 segundos. Para 3 e 4 VANTs, os tempos de execução crescem progressivamente, com médias variando entre 500 e 1000 segundos e maior dispersão. A configuração com 5 VANTs registra o maior custo computacional, com picos superiores a 2000 segundos e comportamento altamente oscilatório, o que reflete o impacto da coordenação entre múltiplos agentes e da expansão do espaço de ações sobre o tempo de simulação.

Perda Acumulada por Episódio: O gráfico central da Figura 5.4 mostra um comportamento estável porém elevado nas curvas de perda acumulada. A configuração com 4 VANTs apresenta a menor perda média ao longo dos episódios, sugerindo bom equilíbrio entre capacidade de exploração e estabilidade do aprendizado. Por outro lado, a configuração com 3 VANTs mostra crescimento acentuado da perda a partir do episódio 100, ultrapassando as demais configurações em termos de variabilidade e valores médios. As configurações com 2 e 5 VANTs apresentam valores intermediários, com curvas mais estáveis após o episódio 150, embora em níveis relativamente elevados. Tais oscilações podem ser atribuídas à complexidade do espaço de estados e à dificuldade de convergência sob diferentes graus de liberdade no posicionamento dos VANTs.

Recompensa Acumulada por Episódio: O gráfico à direita da Figura 5.4 evidencia que todas as configurações evoluem positivamente ao longo dos episódios, atingindo recompensas acumuladas próximas da convergência após o episódio 100. O melhor desempenho é observado na configuração com 5 VANTs, que alcança recompensa média superior a 0,80, refletindo maior cobertura e capacidade de alocação eficiente dos recursos. Em

seguida, as configurações com 4 e 3 **VANTs** apresentam desempenho semelhante, com recompensas médias entre 0,75 e 0,77, demonstrando boa eficiência na política aprendida. A configuração com 2 **VANTs**, embora mais leve computacionalmente, estabiliza em torno de 0,72, evidenciando as limitações de cobertura e conectividade associadas ao menor número de agentes.

Síntese dos Resultados: Os resultados demonstram que o algoritmo **SR-DQN** é capaz de aprender políticas eficazes de posicionamento dinâmico de **VANTs**, mesmo em ambientes com crescente complexidade. O aumento no número de agentes contribui para uma maior recompensa acumulada, mas também implica em maior tempo de execução e variações mais acentuadas na perda acumulada, especialmente nas configurações com 3 e 5 **VANTs**.

A configuração com 5 **VANTs** apresentou a maior recompensa média, indicando elevado potencial de cobertura e conectividade. No entanto, também exibiu o maior custo computacional e oscilações significativas durante o treinamento. Por outro lado, a configuração com 4 **VANTs** mostrou desempenho bastante consistente, com boa recompensa e menor instabilidade. A configuração com 3 **VANTs**, apesar de apresentar a maior perda média, demonstrou comportamento de aprendizado estável. A configuração com 2 **VANTs**, embora computacionalmente mais leve, apresentou limitações em termos de desempenho da política.

Tabela 5.6: Indicadores médios por quantidade de **VANTs** após 300 episódios para o algoritmo **SR-DQN**. [Criada pelo Autor.]

Gateways	Tempo Médio (s)	Perda Média	Recompensa Média
2	422,6	763.489	0,712
3	589,2	650.875	0,735
4	812,9	707.412	0,752
5	1047,3	688.906	0,796

A Tabela 5.6 resume os principais indicadores observados após 300 episódios para cada configuração avaliada. A tabela reforça a conclusão de que a escolha da configuração ideal depende do compromisso entre desempenho e custo. Em ambientes com restrições de tempo, a configuração com 3 **VANTs** pode ser mais apropriada por apresentar boa recompensa com perda relativamente controlada. Em contrapartida, se a prioridade for maximizar a recompensa e a capacidade de cobertura, mesmo com maior tempo de execução, a configuração com 5 **VANTs** se destaca como a melhor opção.

Análise de Desempenho do Treinamento do Algoritmo DA-DDDQN

Nesta seção, são apresentados os resultados obtidos com a aplicação do algoritmo DA-DDDQN ao problema de posicionamento de VANTs atuando como gateways LoRaWAN. O ambiente simulado foi configurado com 121 posições candidatas (11×11) e 50 LoRa-EDs distribuídos. Foram avaliadas quatro configurações distintas quanto ao número de VANTs (2, 3, 4 e 5), todas submetidas a 300 episódios de treinamento.

A Figura 5.5 apresenta os gráficos resultantes com os indicadores de tempo de execução por episódio, perda acumulada e recompensa acumulada.

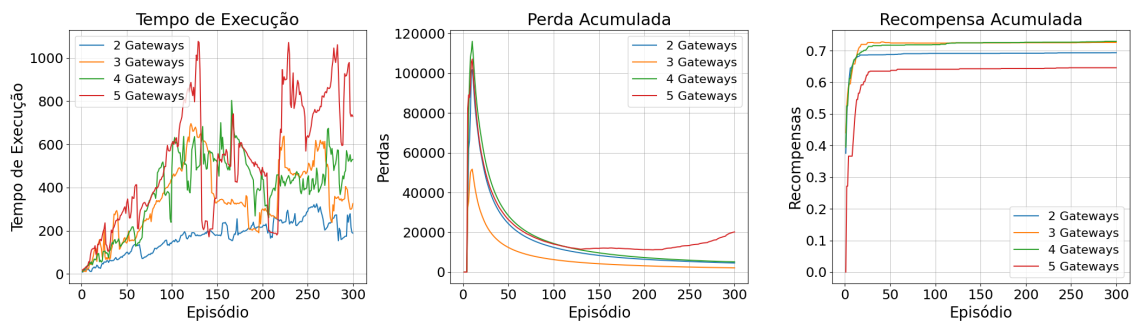


Figura 5.5: Desempenho do DA-DDDQN para 2, 3, 4 e 5 VANTs.
[Criada pelo Autor.]

Tempo de Execução por Episódio: O gráfico à esquerda da Figura 5.5 mostra que o tempo de execução por episódio aumentou significativamente com o número de VANTs, refletindo a complexidade crescente do espaço de ações. Para 2 VANTs, o tempo médio manteve-se em torno de 200 segundos, com baixa variabilidade e tendência de crescimento suave ao longo do treinamento.

Para 3 e 4 VANTs, observaram-se tempos médios em torno de 350 e 500 segundos, respectivamente, com flutuações mais pronunciadas. A configuração com 5 VANTs foi a mais onerosa em termos computacionais, com média superior a 700 segundos e picos recorrentes acima de 1.000 segundos, especialmente após o episódio 150. Esse comportamento evidencia que a sobrecarga computacional associada à coordenação entre múltiplos agentes pode impactar diretamente a escalabilidade do algoritmo.

Perda Acumulada por Episódio: O gráfico central da Figura 5.5 demonstra que todas as configurações iniciaram com perdas elevadas, seguidas de uma rápida convergência nos primeiros 50 episódios. No entanto, o padrão de estabilização variou conforme o número de VANTs.

A configuração com 3 VANTs apresentou a melhor curva de perda, com estabilização precoce e valores consistentemente baixos ao longo dos episódios, destacando-se com a menor perda acumulada entre os cenários. Já a configuração com 2 VANTs, embora

estável, manteve perdas médias mais elevadas, refletindo a limitação da cobertura espacial. A configuração com 4 VANTs teve desempenho intermediário, com boa estabilidade após o episódio 100, enquanto a configuração com 5 VANTs demonstrou comportamento irregular, com aumento das perdas nos episódios finais, indicando possível sobreajuste ou dificuldade de generalização em cenários altamente dinâmicos.

Recompensa Acumulada por Episódio: O gráfico à direita da Figura 5.5 revela que todas as configurações atingiram convergência de recompensa acumulada antes do episódio 100. A configuração com 3 VANTs apresentou a maior recompensa média ao final do treinamento, ultrapassando 0,74, seguida de perto pela configuração com 2 VANTs, que manteve desempenho consistente ao redor de 0,72.

A configuração com 4 VANTs, apesar de apresentar boa estabilidade, ficou ligeiramente abaixo, com média de 0,71. Por outro lado, a configuração com 5 VANTs obteve a menor recompensa acumulada ($< 0,68$), o que pode ser atribuído à maior complexidade de coordenação e ao risco elevado de sobreposição e conflitos entre agentes. Esses resultados indicam que há um ponto ótimo de complexidade para o qual o algoritmo consegue extrair maior benefício da mobilidade sem comprometer a eficiência do processo decisório.

Síntese dos Resultados: Os resultados indicam que a configuração com 2 VANTs apresentou o menor tempo de execução médio e uma boa convergência da política, com recompensa acumulada estável e razoável. A configuração com 3 VANTs destacou-se pela menor perda acumulada ao final do treinamento, associada a uma recompensa levemente superior em relação ao cenário com 2 unidades. O cenário com 4 VANTs atingiu a maior recompensa acumulada dentre todas as configurações, evidenciando um aprendizado eficaz. No entanto, isso implicou em um custo computacional significativamente mais elevado. Em contraste, o uso de 5 VANTs, apesar de sua potencial capacidade de cobertura, resultou em elevada variabilidade no tempo de execução, aumento expressivo da perda acumulada e recompensa inferior às demais configurações. Esse comportamento sugere que a complexidade adicional imposta pela coordenação entre múltiplos agentes pode comprometer o desempenho global da política aprendida.

A Tabela 5.7 apresenta um resumo quantitativo dos principais indicadores observados ao final de 300 episódios de simulação para cada configuração de número de VANT.

Tabela 5.7: Indicadores médios por quantidade de VANTs após 300 episódios para o algoritmo DA-DDDQN. [Criada pelo Autor.]

VANTs	Tempo Médio (s)	Perda Média	Recompensa Acumulada Média
2	210,5	13.430	0,712
3	384,2	10.230	0,738
4	527,8	11.780	0,756
5	730,9	20.540	0,685

A Tabela 5.7 reforça que a configuração com 4 VANTs apresentou o melhor equilíbrio entre desempenho e robustez da política, ao custo de maior complexidade computacional. Já o uso de 5 VANTs, apesar de intuitivamente promissor, não ofereceu ganhos proporcionais e implicou desafios substanciais de coordenação e estabilidade, sugerindo a existência de um ponto ótimo entre cobertura e eficiência computacional.

Análise de Desempenho do Treinamento do Algoritmo NSE-A2C

O algoritmo NSE-A2C foi avaliado quanto à sua capacidade de aprender políticas eficientes de alocação dinâmica de VANTs em redes IoT não-3GPP, considerando diferentes configurações de rede. A investigação abrangeu dois principais aspectos: o impacto da variação no número de VANTs e a influência da quantidade de dispositivos na rede. Os experimentos foram conduzidos em ambiente ns-3, com 144 posições candidatas para alocação, 200 dispositivos, e variações de 2 a 5 VANTs para o primeiro aspecto, além de cenários com 50, 100 e 200 dispositivos com 3 VANTs no segundo. Foram analisadas as métricas de tempo médio de execução por episódio, perdas médias do ator e do crítico, e recompensa acumulada média, ao longo de 300 episódios. Os resultados são apresentados nas Figuras 5.6 e 5.7.

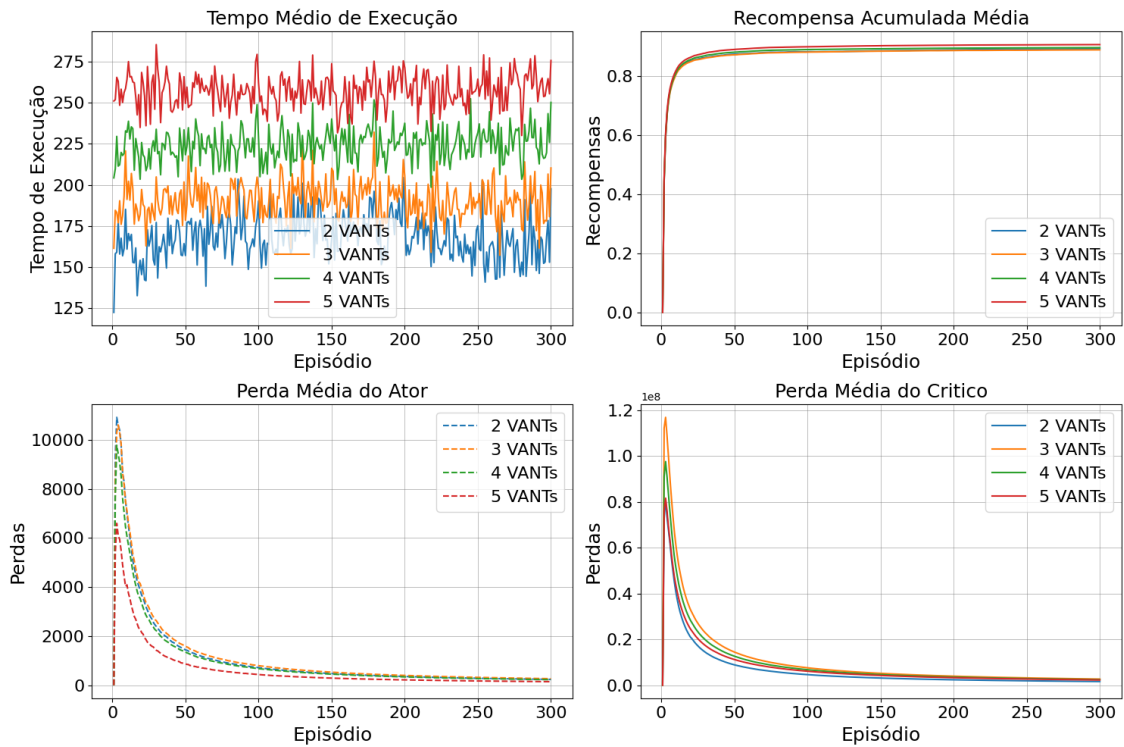


Figura 5.6: Impacto da Variação no Número de VANTs no Desempenho do NSE-A2C. [Criada pelo Autor.]

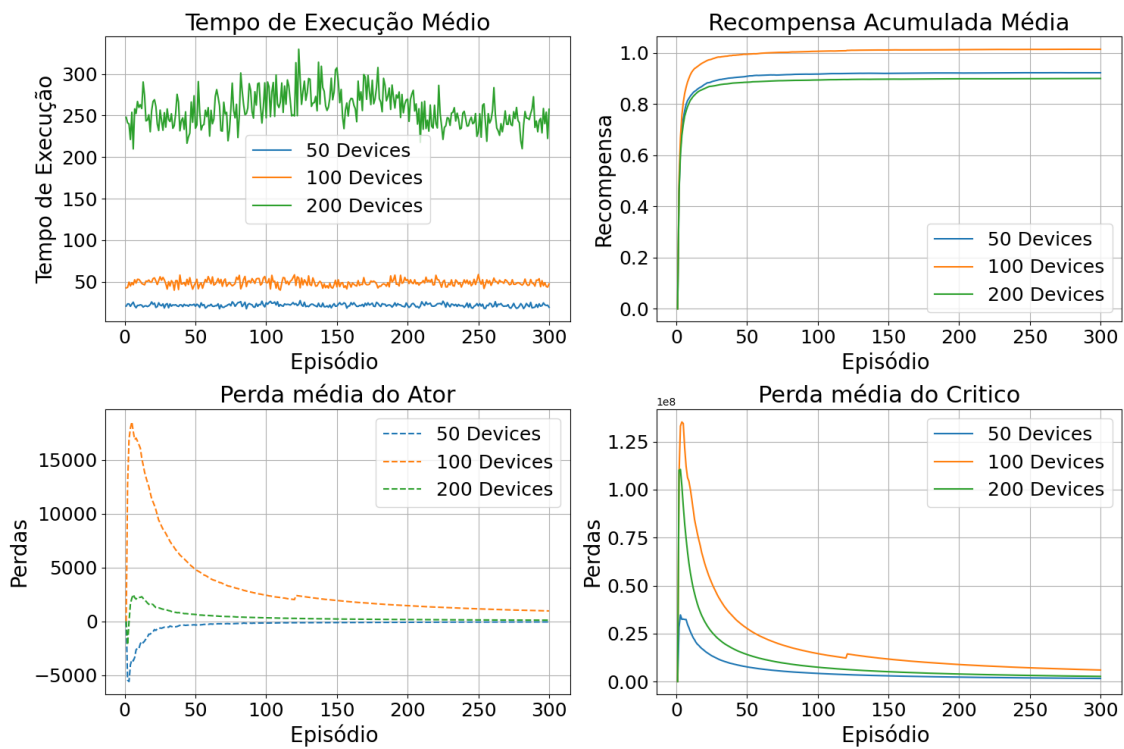


Figura 5.7: Impacto da Variação na Quantidade de Dispositivos no Desempenho do NSE-A2C. [Criada pelo Autor.]

Tempo médio de execução. Nos dois conjuntos de experimentos, observou-se que o tempo de execução por episódio aumentou proporcionalmente à complexidade da configuração. Na análise com variação do número de VANTs, Figura 5.6, os tempos médios cresceram com o acréscimo de agentes. O cenário com 2 VANTs apresentou os menores tempos, enquanto o com 5 VANTs registrou os maiores, com picos próximos de 275 segundos. Esse crescimento está relacionado à ampliação do espaço de ações e ao esforço computacional exigido para coordenar múltiplos agentes móveis simultaneamente.

Na análise com diferentes quantidades de dispositivos, Figura 5.7, o tempo médio de execução também se elevou significativamente. Para 50 dispositivos, os tempos permaneceram abaixo de 30 segundos. No cenário com 100 dispositivos, variaram entre 50 e 70 segundos. Já para 200 dispositivos, os tempos ultrapassaram 300 segundos, refletindo a intensificação dos eventos simulados e o aumento da carga computacional no simulador ns-3.

Perda média do ator. A perda média do ator apresentou queda progressiva e estável em todos os experimentos, indicando aprendizado consistente da política. Na variação do número de VANTs, Figura 5.6, as configurações com mais agentes exibiram perdas iniciais menores e curvas de convergência mais suaves. O caso com 2 VANTs iniciou com perdas próximas de 12.000, enquanto as perdas para os experimentos com 5 VANTs começou ao redor de 6.000, todos convergindo para valores próximos de zero.

No experimento com variação na quantidade de dispositivos, Figura 5.7, observou-se padrão semelhante: quanto maior a carga, mais estável foi a curva de perda. O cenário com 50 dispositivos apresentou perdas iniciais abaixo de -5.000 , enquanto o com 200 dispositivos apresentou perdas acima de 15000, ambos mostraram convergências graduais e suaves, sugerindo que a diversidade de experiências de treinamento contribuiu para a generalização da política.

Perda média do crítico. A perda do crítico, nos diferentes cenários, apresentou valores iniciais elevados, seguidos por decaimento contínuo ao longo dos episódios. Na variação do número de VANTs, Figura 5.6, a perda inicial dos cenários experimentados, i.e., 2, 3, 4 e 5 VANTs iniciaram o treinamento com perdas entre $0,8 \times 10^8$ e $1,2 \times 10^8$. As curvas de todas as configurações estabilizaram-se a partir do episódio 150, evidenciando a capacidade do algoritmo em estimar os valores de retorno mesmo em espaços de estado mais amplos.

No experimento com diferentes quantidades de dispositivos, Figura 5.7, o cenário com 100 dispositivos apresentou a maior perda inicial, ultrapassando $1,25 \times 10^8$. As três curvas convergiram para níveis similares e próximos a zero após o episódio 200, indicando maior estabilidade na estimativa da função de valor.

Recompensa acumulada média. As curvas de recompensa acumulada mostraram rápida convergência em todos os cenários, com estabilização após os 50 primeiros episódios. Na análise da variação do número de VANTs, Figura 5.6, o melhor desempenho foi observado com 5 VANTs, que atingiu recompensas médias superiores a 0,96. O cenário com 2 VANTs estabilizou-se em torno de 0,91, refletindo redução de desempenho relacionada à limitação de cobertura.

Quanto à variação na quantidade de dispositivos, Figura 5.7, o cenário com 100 dispositivos alcançou a maior recompensa acumulada média (cerca de 1,05). O caso com 50 dispositivos permaneceu ligeiramente abaixo de 0,91, e o com 200 estabilizou-se em torno de 0,93. Esses resultados sugerem que a presença de carga moderada favorece a aprendizagem de políticas mais eficazes.

Síntese dos resultados. O algoritmo NSE-A2C demonstrou comportamento consistente e capacidade de generalização diante das variações na quantidade de VANTs e na densidade de dispositivos da rede. As curvas de recompensa acumulada evidenciaram convergência rápida em todos os cenários, com estabilidade mantida ao longo dos episódios. Apesar de aumentos no custo computacional com a complexidade do ambiente, os níveis de perda dos agentes mantiveram-se sob controle, com decaimento progressivo. As Tabelas 5.8 e 5.9 apresentam a síntese quantitativa dos principais resultados obtidos com base nas curvas médias.

Tabela 5.8: *Resumo quantitativo do desempenho do NSE-A2C com variação no número de VANTs [Valores estimados a partir das curvas médias]. [Criada pelo Autor.]*

Métrica	2 VANTs	3 VANTs	4 VANTs	5 VANTs
Tempo médio de execução (s)	160	180	220	260
Perda inicial do ator	11.000	10.000	8.500	7.500
Perda inicial do crítico	$1,1 \times 10^8$	$1,0 \times 10^8$	$8,5 \times 10^7$	$8,0 \times 10^7$
Recompensa acumulada média	0,90	0,91	0,92	0,93

Tabela 5.9: *Resumo dos resultados para diferentes quantidades de dispositivos com NSE-A2C. [Criada pelo Autor.]*

Métrica	50 Devices	100 Devices	200 Devices
Tempo médio de execução	~ 30s	~60s	>250s
Perda média do ator	-4.500	-2.000	-1.000
Perda média do crítico	$7,0 \times 10^7$	$9,0 \times 10^7$	$5,0 \times 10^7$
Recompensa acumulada	~ 0,90	~ 1,00	~0,93

Análise de Desempenho do Treinamento do Algoritmo RG2E-PPO

Nesta seção, avalia-se o desempenho do algoritmo RG2E-PPO, mantendo-se as configurações experimentais previamente adotadas nas análises do algoritmo NSE-A2C, a fim de permitir comparações consistentes. A análise contempla a variação no número de VANTs e na quantidade de dispositivos, com foco nas métricas de tempo médio de execução por episódio, perdas médias do ator e do crítico, e recompensa acumulada média ao longo de 300 episódios. Os resultados são apresentados nas Figuras 5.8 e 5.9.

Tempo médio de execução. Nos dois experimentos, o tempo de execução apresentou tendência de estabilização após os primeiros 50 episódios, com redução das flutuações à medida que o algoritmo convergia. Na variação do número de VANTs, Figura 5.8, observou-se crescimento do tempo médio conforme aumentava a quantidade de agentes, com destaque para o cenário com 5 VANTs com tempo de execução alcançando tempos superiores a 80s nos primeiros episódios diminuindo para valores próximo de 40s com o avanço do episódios. Esse comportamento está relacionado à maior complexidade do espaço de ações e à demanda computacional para simulações mais detalhadas associadas ao maior número de VANTs do cenário experimentado.

Por outro lado, a variação na quantidade de dispositivos impactou de forma ainda mais pronunciada. O tempo médio para 50 dispositivos foi inferior a 30 segundos, enquanto o cenário com 200 dispositivos ultrapassou 250 segundos nos primeiros episódios e estabilizado em valores acima dos 100 segundos. Esse crescimento reflete o aumento das interações e da carga de comunicação no ns-3 relacionadas ao maior número de dispositivos simulados.

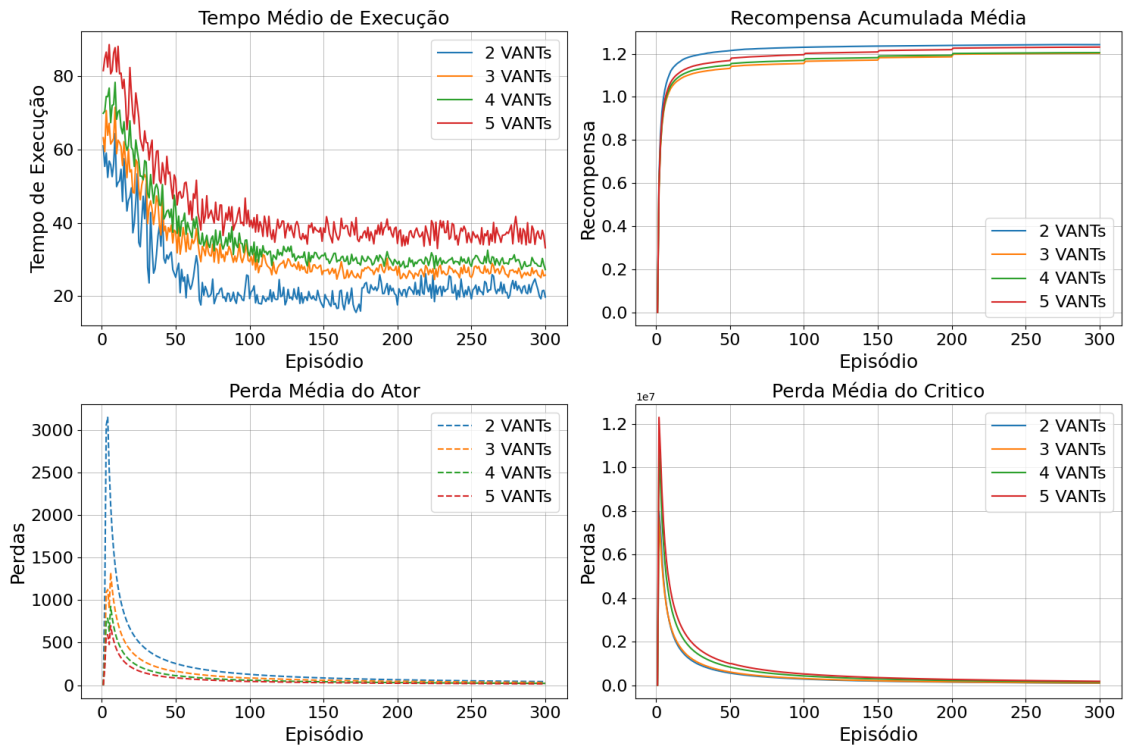


Figura 5.8: Impacto da variação no número de VANTs no desempenho do RG2E-PPO. [Criada pelo Autor.]

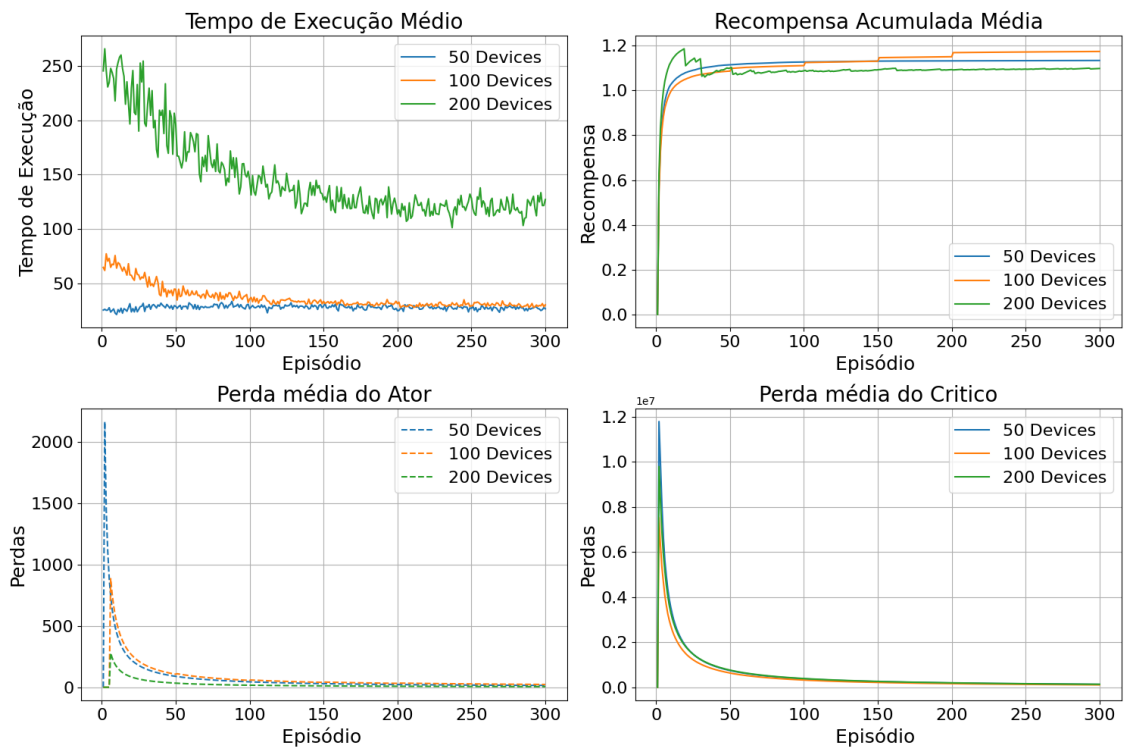


Figura 5.9: Impacto da variação na quantidade de dispositivos no desempenho do RG2E-PPO. [Criada pelo Autor.]

Perda média do ator. A perda média do ator apresentou comportamento decrescente e estável em todos os cenários, com convergência próxima de zero. Na variação de **VANTs**, o cenário com 2 **VANTs** iniciou com perdas superiores a 2.000, enquanto os cenários com 3, 4 e 5 **VANTs** apresentaram perdas iniciais menores e decaimento mais rápido. Esses resultados indicam que a maior quantidade de agentes móveis contribui para maior flexibilidade no aprendizado da política, favorecendo a exploração e a convergência.

Quanto à variação na carga da rede, observou-se padrão semelhante: cenários com maior número de dispositivos apresentaram perdas iniciais menores e curvas de convergência mais suaves. Essa tendência sugere que ambientes com maior diversidade de interações favorecem o processo de generalização da política.

Perda média do crítico. A perda do crítico iniciou-se com valores elevados em todos os cenários, especialmente na presença de maior número de **VANTs** ou dispositivos. No entanto, observou-se decaimento exponencial em todos os casos, com estabilização por volta do episódio 150. No cenário com 5 **VANTs**, a perda inicial ultrapassou $1,2 \times 10^7$, enquanto nas demais configurações os valores foram progressivamente menores. Esse comportamento reforça que ambientes mais complexos demandam maior esforço para a estimação da função de valor pelo componente crítico da rede.

Na análise com variação na densidade de dispositivos, os cenários experimentados apresentaram perdas iniciais entre $0,8 \times 10^7$ e $1,2 \times 10^7$. As curvas de perda convergiram para valores próximos a zero, evidenciando a capacidade de generalização da política nos cenários experimentados.

Recompensa acumulada média. As curvas de recompensa acumulada evidenciaram que o algoritmo **RG2E-PPO** aprendeu políticas eficazes de forma rápida e estável. Em todos os cenários, a convergência foi observada até aproximadamente o episódio 50, com posterior estabilização em níveis mais elevados. Na análise com variação do número de **VANTs**, o melhor desempenho foi obtido no cenário com 2 **VANTs**, cuja recompensa média ultrapassou 1,23. Os demais cenários, 3, 4 e 5 **VANTs**, apresentaram valores finais próximos a 1,20, com pequenas variações entre eles. Este resultado sugere que, embora o aumento no número de agentes amplie a capacidade de cobertura, a complexidade adicional no espaço de ações pode dificultar o refinamento da política ótima.

Na análise por quantidade de dispositivos, o melhor resultado foi alcançado novamente com 50 dispositivos, atingindo uma recompensa média superior a 1,15. O cenário com 100 dispositivos apresentou desempenho similar, com valores ligeiramente abaixo, enquanto o caso com 200 dispositivos estabilizou-se em torno de 1,10. Essa tendência reforça a hipótese de que há um ponto de equilíbrio entre a carga da rede e a capacidade dos **VANTs** em proporcionar cobertura eficiente. Em cargas mais elevadas,

a saturação de recursos pode limitar os ganhos em desempenho, mesmo com políticas bem ajustadas.

Síntese dos resultados. O algoritmo **RG2E-PPO** apresentou desempenho consistente e robusto diante de diferentes configurações da rede, destacando-se por sua capacidade de adaptação, estabilidade durante o treinamento e boa generalização frente à variação tanto no número de **VANTs** quanto na quantidade de dispositivos IoT. A Tabela 5.10 resume os principais resultados obtidos com a variação do número de **VANTs**, enquanto a Tabela 5.11 apresenta os resultados da análise sob diferentes densidades de dispositivos.

Tabela 5.10: *Resumo quantitativo do desempenho do RG2E-PPO com variação no número de VANTs. Valores estimados a partir das curvas médias. [Criada pelo Autor.]*

Métrica	2 VANTs	3 VANTs	4 VANTs	5 VANTs
Tempo médio de execução (s)	20	26	31	38
Perda inicial do ator	3.200	1.400	800	600
Perda inicial do crítico	$1,2 \times 10^7$	$9,5 \times 10^6$	$7,5 \times 10^6$	$6,5 \times 10^6$
Recompensa acumulada média	1,24	1,22	1,21	1,19

Tabela 5.11: *Resumo dos resultados para diferentes quantidades de dispositivos com RG2E-PPO. [Criada pelo Autor.]*

Métrica	50 Devices	100 Devices	200 Devices
Tempo médio de execução	~30s	~55s	>120s
Perda média do ator	2.200	1.000	400
Perda média do crítico	$1,15 \times 10^7$	$8,0 \times 10^6$	$6,8 \times 10^6$
Recompensa acumulada	~1,18	~1,15	~1,12

Os resultados obtidos reforçam que o **RG2E-PPO** é uma abordagem promissora para o problema de alocação dinâmica de **VANTs** em redes **IoT** não-**3GPP**. A capacidade de limitar variações abruptas na política, característica inerente ao método PPO, contribuiu significativamente para a convergência eficiente e para a estabilidade observada ao longo dos episódios de treinamento, mesmo em cenários com alta complexidade computacional ou elevada densidade de dispositivos. Além disso, a evolução suave das curvas de perda e a estabilidade das recompensas indicam que o modelo alcança um bom equilíbrio entre exploração e exploração, com elevada qualidade nas políticas aprendidas.

Análise Comparativa dos Tempos Médios de Treinamento dos Algoritmos

A Figura 5.10 apresenta a evolução do tempo médio de execução dos algoritmos em treinamento **SR-DQN**, **DA-DDDQN**, **NSE-A2C** e **RG2E-PPO**, considerando diferentes quantidades de **VANTs**, e.g., 2 e 4. O tempo médio de execução foi avaliado ao longo de 100 episódios, refletindo o impacto da escolha do algoritmo na eficiência computacional e na escalabilidade da solução.

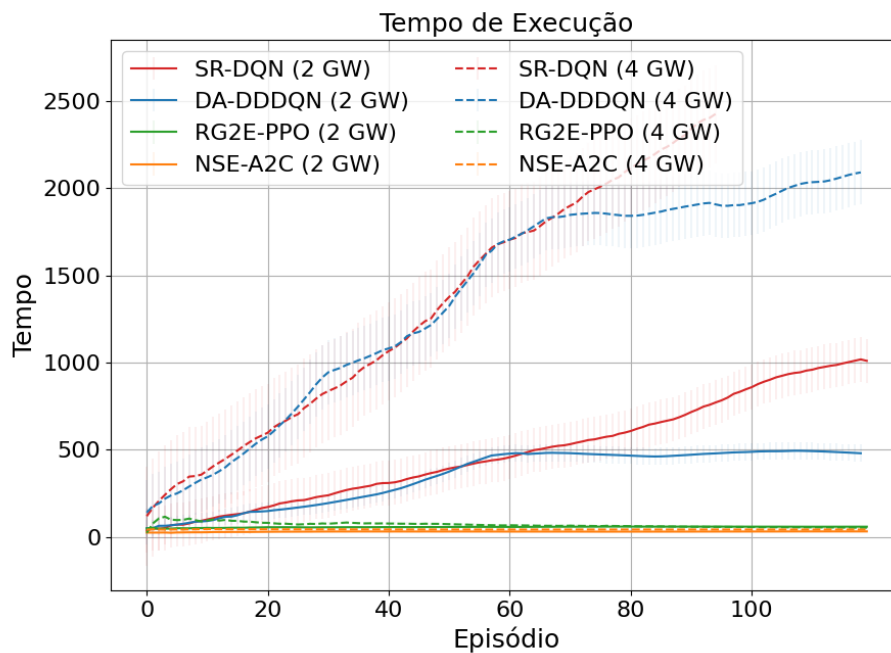


Figura 5.10: Variações nos tempos médios de execução dos algoritmos para 2 e 4 VANTs e 100 dispositivos. [Criada pelo Autor.]

Comportamento dos algoritmos baseados em DQN: Os métodos **SR-DQN** e **DA-DDDQN** apresentaram tempos médios de execução significativamente mais elevados em comparação com os algoritmos baseados em política, como o **NSE-A2C** e o **RG2E-PPO**. Observou-se uma tendência de crescimento acentuado do tempo de execução ao longo dos episódios, evidenciando aumento progressivo da complexidade computacional durante o treinamento. Essa característica foi particularmente pronunciada nos cenários com 4 **VANTs**, nos quais os tempos de execução ultrapassaram 1.200 segundos por episódio. Esse comportamento pode ser atribuído a fatores estruturais dos algoritmos baseados em função de valor, como o uso de memória de *replay* com amostragem aleatória, a necessidade de atualização periódica da rede-alvo e o crescimento exponencial do espaço de ações conforme o número de **VANTs** aumenta. Esses fatores impactam direta-

mente o custo computacional por episódio, tornando esses métodos menos escaláveis em ambientes com alta dimensionalidade de estados e ações.

Comportamento dos algoritmos baseados em gradiente de política: Os métodos [NSE-A2C](#) e [RG2E-PPO](#) apresentaram tempos de execução inferiores e mais estáveis ao longo dos episódios, quando comparados às abordagens baseadas em função de valor. O algoritmo [NSE-A2C](#) apresentou os menores tempos médios, com crescimento praticamente desprezível mesmo nos cenários mais complexos. O algoritmo [RG2E-PPO](#) apresentou comportamento semelhante, com leve aumento no tempo médio, porém mantendo estabilidade. Essa eficiência computacional pode ser atribuída à ausência de memória de *replay* e à realização de atualizações diretas a partir de transições recentes. Além disso, a ausência de múltiplas inferências por ação e de sincronização periódica com uma rede-alvo favorece o paralelismo e a escalabilidade dos métodos baseados em política em ambientes com espaços de ação mais amplos.

Impacto do número de VANTs: A presença de quatro [VANTs](#) acentuou as diferenças de desempenho entre os algoritmos. Nos métodos [SR-DQN](#) e [DA-DDDQN](#), observou-se crescimento expressivo no tempo médio de execução, evidenciando limitações na escalabilidade computacional. Por outro lado, nos métodos [NSE-A2C](#) e [RG2E-PPO](#), o impacto da variação no número de agentes foi consideravelmente menor, o que sugere maior capacidade de adaptação a cenários com múltiplos [VANTs](#).

Síntese dos resultados: Os algoritmos [SR-DQN](#) e [DA-DDDQN](#) apresentaram tempos de execução mais elevados, especialmente em cenários com maior número de [VANTs](#), o que pode comprometer sua viabilidade em aplicações com restrições de tempo ou com necessidade de operação em tempo real. Tal comportamento está associado ao uso de estruturas como memória de *replay* e redes-alvo, além da exigência de múltiplas inferências sobre o espaço de ações. Em contraste, os algoritmos baseados em gradiente de política, [NSE-A2C](#) e [RG2E-PPO](#), demonstraram desempenho computacional mais eficiente, com maior estabilidade ao longo do treinamento. Esses métodos mostraram-se mais escaláveis em ambientes com elevada dimensionalidade de ações e dispensam o armazenamento extensivo de transições passadas, o que reduz a sobrecarga computacional. Os resultados indicam que, para problemas de alocação dinâmica de [VANTs](#) em redes [IoT](#), em que há restrições computacionais e necessidade de adaptação contínua, os métodos baseados em política representam alternativas mais eficazes e robustas que os algoritmos tradicionais baseados em função de valor.

Resultados da Etapa de Avaliação

A seguir, são apresentados os resultados da aplicação política π^* aprendida por cada agente em cenários de avaliação distintos, baseados em um conjunto de sementes ϕ não utilizadas durante o treinamento.

Análise Comparativa de PDR entre os Algoritmos de DRL

A métrica de **PDR** é um dos principais indicadores de **QoS** em aplicações de redes **IoT** com suporte de mobilidade. A Figura 5.11 apresenta a comparação do **PDR** entre os algoritmos **NSE-A2C**, **RG2E-PPO**, **SR-DQN** e **DA-DDDQN**, em ambiente de simulação **ns-3**. Os experimentos foram realizados com 2 e 4 **VANTs**, variando o número de posições candidatas ($v \in \{49, 64, 81, 100, 121, 144\}$) e de dispositivos **LoRa-EDs** ($d \in \{50, 100, 150, 200\}$).

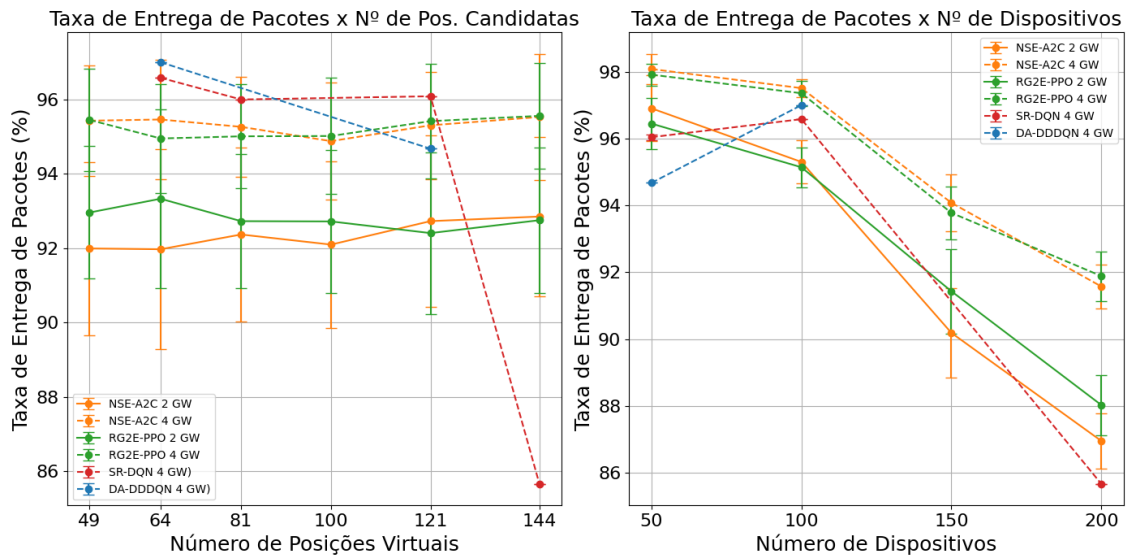


Figura 5.11: PDR para diferentes algoritmos de DRL em função do número de posições candidatas (à esquerda) e do número de dispositivos (à direita). [Criada pelo Autor.]

Entre os algoritmos avaliados, o **RG2E-PPO** apresentou o melhor desempenho global em termos de **PDR**. Em todos os cenários, a política aprendida manteve a taxa de entrega acima de 94%, mesmo com o aumento da densidade de dispositivos ou da granularidade espacial. Esse comportamento está associado à combinação entre vantagem generalizada (**GAE**), regularização por entropia e controle de atualização da política via *clipping*, características que proporcionam maior estabilidade e capacidade de adaptação dinâmica.

O algoritmo **NSE-A2C** apresentou desempenho competitivo, embora com sensibilidade maior à variação na carga da rede. Em especial, para valores de $d \geq 150$,

observou-se uma redução do **PDR** médio para níveis abaixo de 90%. Esse comportamento pode ser atribuído à maior suscetibilidade da política estocástica às penalizações impostas por colisões e evasões de cobertura. Ainda assim, o **NSE-A2C** manteve resultados estáveis nas demais configurações e se destacou pelo baixo custo computacional, tornando-se uma opção atrativa em ambientes com restrições de recursos.

Os algoritmos baseados em função de valor, e.g., **SR-DQN** e **DA-DDDQN** apresentaram bom desempenho em pontos específicos da curva, com **PDR** superior a 96% em cenários iniciais com menor número de posições candidatas ($v = 49$ a $v = 121$). No entanto, ambos enfrentaram limitações severas de execução nos cenários mais complexos. Para o **SR-DQN**, a queda de desempenho em $v = 144$, com **PDR** inferior a 86%, não pôde ser devidamente investigada, pois a execução foi interrompida devido ao alto custo computacional associado à manutenção de múltiplas redes neurais, atualizações concorrentes e *buffers* de experiência extensos.

De forma similar, o algoritmo **DA-DDDQN**, embora tenha adotado avanços como estimativas duplas e decomposição de política entre agentes, apresentou dificuldades operacionais causadas pela alta latência nas atualizações e pelo consumo de memória. Como resultado, ambos os algoritmos tiveram seus experimentos limitados a configurações parciais, sem geração completa de curvas de avaliação.

Tabela 5.12: *Resumo das limitações computacionais dos algoritmos baseados em função de valor e seus impactos na avaliação. [Desenvolvido pelo Autor.]*

Algoritmo	Componentes críticos de alta demanda	Limitações enfrentadas	Impactos na avaliação
SR-DQN	<ul style="list-style-type: none"> • <i>Replay buffer</i> com amostragem prioritária • Múltiplas redes neurais (principal e alvo) • Atualizações concorrentes por episódio 	<ul style="list-style-type: none"> • Alto consumo de memória • Longo tempo de processamento por episódio • Requisitos de hardware especializados 	<ul style="list-style-type: none"> • Curvas de avaliação incompletas • Interrupção da execução em cenários complexos ($v = 144$) • Dificuldade de replicação sob infraestrutura convencional
DA-DDDQN	<ul style="list-style-type: none"> • Dupla estimativa com redes separadas • Arquitetura <i>Dueling</i> • Decomposição por agentes 	<ul style="list-style-type: none"> • Alta latência nas atualizações • Escalabilidade limitada em ambientes densos • Exigência elevada de sincronização 	<ul style="list-style-type: none"> • Execução interrompida em múltiplos cenários • Resultados limitados a pontos específicos • Ausência de curvas contínuas de desempenho

A Tabela 5.12 resume os principais componentes computacionais desses dois

métodos, suas limitações práticas e os impactos observados nos experimentos. Em especial, destaca-se que os efeitos dessas limitações transcendem o tempo de execução, afetando diretamente a completude, a reprodutibilidade e a comparabilidade dos resultados com os demais algoritmos.

Dessa forma, os dados obtidos indicam que, sob as restrições de infraestrutura disponíveis, o **RG2E-PPO** representa a abordagem mais robusta para o problema de posicionamento dinâmico de **VANTs**, apresentando elevada eficácia em termos de **PDR** mesmo em cenários de alta complexidade. O **NSE-A2C** consolidou-se como uma alternativa viável e eficiente, especialmente em aplicações que exigem equilíbrio entre desempenho e simplicidade computacional. Por outro lado, **SR-DQN** e **DA-DDDQN**, embora promissores do ponto de vista teórico, demandam plataformas computacionais mais robustas para possibilitar sua execução completa e permitir análises comparativas consistentes.

Análise comparativa das métricas de QoS para os algoritmos NSE-A2C e RG2E-PPO

A avaliação da **QoS** constitui um elemento central para a verificação da eficácia dos algoritmos de **DRL** aplicados à alocação dinâmica de recursos em redes **IoT** com suporte de **VANTs**. A Figura 5.12 apresenta a evolução da média global de **QoS** ao longo dos episódios de treinamento para os algoritmos **NSE-A2C** e **RG2E-PPO**, considerando diferentes configurações no ambiente simulado com o *ns-3*.

Os resultados indicam que o algoritmo **RG2E-PPO** apresentou os maiores níveis de **QoS**, com convergência rápida e manutenção estável ao longo dos episódios. Essa performance pode ser atribuída à arquitetura baseada em políticas determinísticas, aliada ao uso de limitação de variação de política (*clipping*) e estimação de **GAE**, que favorecem a estabilidade e a eficiência do processo de aprendizagem, mesmo em ambientes com alta densidade de dispositivos e variabilidade topológica.

O algoritmo **NSE-A2C**, por sua vez, também demonstrou evolução consistente da **QoS**, com convergência gradual para valores superiores a 0,75. Sua política estocástica, associada ao controle adaptativo de entropia, favorece a exploração equilibrada do espaço de estados, combinando simplicidade computacional e capacidade adaptativa. Esse comportamento posiciona o **NSE-A2C** como uma alternativa viável para cenários com restrições de *hardware*.

Os algoritmos **SR-DQN** e **DA-DDDQN** não foram incluídos nesta etapa da avaliação devido às limitações práticas relacionadas à execução prolongada dos experimentos. A elevada demanda computacional dessas abordagens, decorrente do uso de múltiplas redes neurais, *replay buffers* de alta capacidade e mecanismos de atualização intensivos, inviabilizou a coleta de dados representativos em tempo hábil em plataformas convencio-

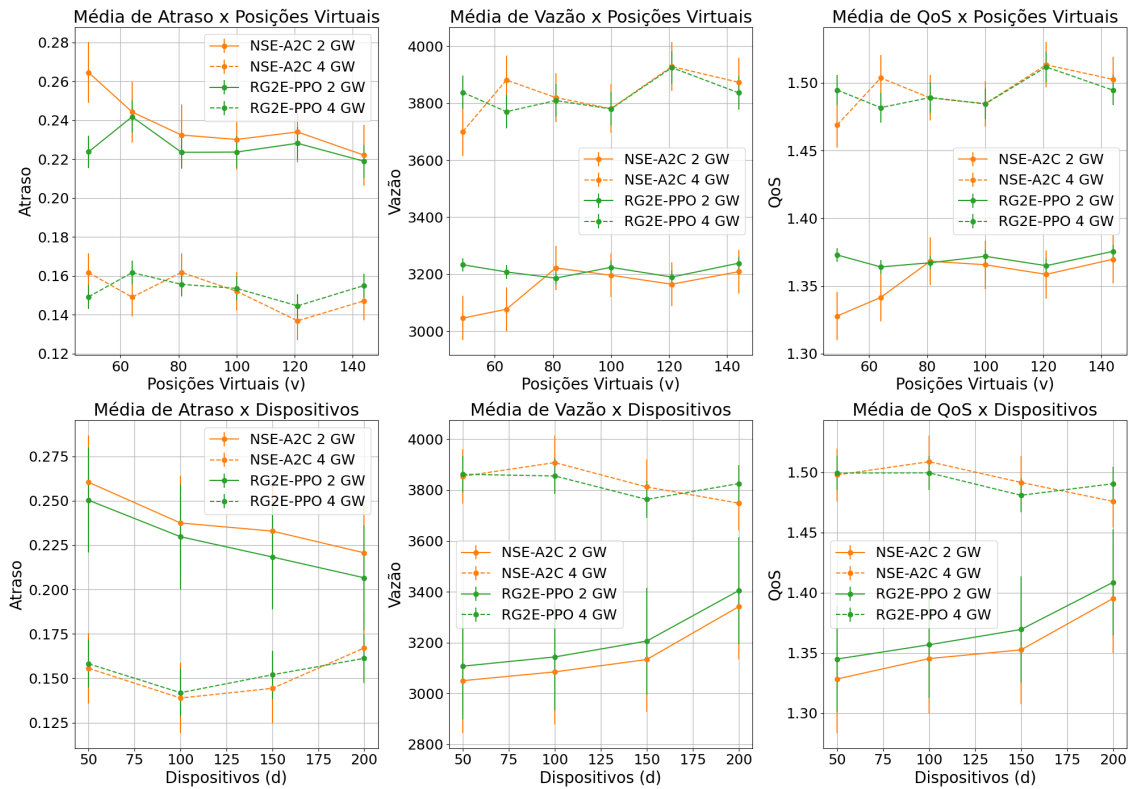


Figura 5.12: Evolução da QoS ao longo dos episódios para os algoritmos NSE-A2C e RG2E-PPO. [Criada pelo Autor.]

nais. Como resultado, não foi possível realizar uma comparação estatisticamente robusta entre esses métodos e os algoritmos baseados em política.

Em síntese, os resultados desta análise confirmam o **RG2E-PPO** como a abordagem mais eficaz no contexto avaliado, destacando-se pela estabilidade e desempenho em termos de **QoS**. O algoritmo **NSE-A2C** revelou-se competitivo e apropriado para aplicações com menor capacidade computacional. A ausência de dados para os algoritmos **SR-DQN** e **DA-DDDQN** reforça a necessidade de infraestrutura computacional mais avançada para suas execuções plenas e avaliação comparativa adequada.

5.5 Conclusão

Este capítulo apresentou uma abordagem baseada em aprendizado por reforço profundo (**DRL**) para a alocação dinâmica de recursos em redes **IoT** com suporte de **VANTs**, estruturada a partir de uma estratégia híbrida que combina a formulação exata do problema com técnicas adaptativas de aprendizado. A formulação inicial, baseada em **MILP**, foi utilizada para gerar soluções estáveis e determinísticas, que serviram como referência para a definição das políticas iniciais dos agentes. A partir desse ponto, os algoritmos de **DRL** foram treinados em ambiente simulado no **ns-3**, permitindo a

adaptação contínua das políticas frente às condições variáveis da topologia da rede e da distribuição dos dispositivos.

Foram implementados e avaliados quatro algoritmos: [SR-DQN](#), [DA-DDDQN](#), [NSE-A2C](#) e [RG2E-PPO](#). Os métodos baseados em função de valor apresentaram maior custo computacional e menor escalabilidade em cenários com múltiplos [VANTs](#), especialmente devido à complexidade associada ao uso de *replay buffers*, redes-alvo e à necessidade de múltiplas inferências sobre o espaço de ações. Por outro lado, os métodos baseados em gradiente de política demonstraram desempenho mais consistente, com tempos de execução reduzidos e maior estabilidade ao longo dos episódios.

O algoritmo [NSE-A2C](#) apresentou desempenho equilibrado, conciliando simplicidade arquitetural e eficácia na aprendizagem da política. Já o [RG2E-PPO](#) destacou-se como a solução mais robusta, com rápida convergência, elevada qualidade das decisões e resiliência frente à complexidade do espaço de estados e ações. Ambos os métodos demonstraram capacidade superior de generalização e adaptação frente à variação no número de dispositivos e de [VANTs](#).

A integração entre a modelagem exata e o aprendizado por reforço configurou-se como uma estratégia eficaz para superar as limitações individuais de cada abordagem. A modelagem determinística proporcionou precisão na definição do problema, enquanto o [DRL](#) assegurou adaptabilidade, escalabilidade e resposta eficiente em tempo de execução. A análise comparativa evidenciou que a abordagem híbrida permite a alocação dinâmica de [VANTs](#) com maior flexibilidade, menor dependência de reconfigurações manuais e melhores indicadores de desempenho, mesmo em ambientes densos e altamente dinâmicos.

Conclui-se, portanto, que a aplicação de algoritmos de [DRL](#), especialmente em sua vertente baseada em políticas, é adequada para a resolução do problema de alocação dinâmica de recursos em redes [IoT](#) com [VANTs](#), oferecendo uma alternativa escalável e eficaz frente às soluções exatas tradicionais.

Considerações Finais e Trabalhos Futuros

Esta tese apresentou uma proposta para a alocação dinâmica de recursos em fatias de redes IoT não-3GPP, utilizando VANTs como elementos móveis e adaptativos da infraestrutura de comunicação. A originalidade do trabalho reside na concepção e implementação de uma abordagem *híbrida* para a alocação dinâmica de recursos em redes IoT não-3GPP. Essa abordagem combina, de forma sinérgica, a precisão dos métodos de otimização exata baseados em MILP, empregados para o posicionamento inicial ótimo dos VANTs, com a flexibilidade adaptativa de algoritmos avançados de DRL, responsáveis pelo reposicionamento dinâmico e autônomo em ambientes variáveis. Tal integração permite explorar simultaneamente as vantagens estruturais de modelos determinísticos, úteis para garantir eficiência e rigor na fase inicial de planejamento, e as capacidades de aprendizado e adaptação dos modelos iterativos, fundamentais para operar em cenários com elevada incerteza e dinamicidade. Essa arquitetura em duas fases amplia a escalabilidade, a robustez e a autonomia da solução, consolidando a proposta como uma alternativa eficaz para a orquestração inteligente de redes móveis não-3GPP sob condições críticas de conectividade.

A proposta metodológica foi estruturada em duas fases complementares. A primeira, de natureza determinística, formulou o problema de posicionamento ótimo dos VANTs e a alocação dos dispositivos IoT em *slices* por meio de modelos MILP, com foco na minimização do número de *gateways* implantados e na maximização da QoS, considerando métricas como cobertura, latência e eficiência espectral. A segunda fase, de caráter adaptativo, utilizou os resultados da etapa exata como ponto de partida para o desenvolvimento de agentes inteligentes capazes de atuar em ambientes de maior granularidade, nos quais a complexidade do espaço de estados inviabiliza soluções determinísticas. Nesse contexto, foram propostos e avaliados quatro algoritmos originais de DRL, e.g., SR-DQN, DA-DDDQN, NSE-A2C e RG2E-PPO, que viabilizaram o reposicionamento autônomo e dinâmico dos VANTs, frente à variabilidade espacial e temporal da rede.

A proposta também respondeu, de maneira clara e fundamentada, às questões de pesquisa delineadas no Capítulo 1.1. A primeira questão, sobre como garantir conectivi-

dade eficiente e QoS em situações de demanda emergencial, foi atendida com a utilização de VANTs como *gateways* móveis integrados à infraestrutura 5G, otimizados por métodos de DRL. Os resultados experimentais demonstraram ganhos significativos na taxa de entrega de pacotes, redução da latência e aumento da cobertura, mesmo em cenários críticos com alta densidade de dispositivos. A segunda questão, que trata da otimização de custos e desempenho, foi solucionada pela combinação MILP-DRL, que reduziu de forma expressiva o número necessário de VANTs, minimizando o CAPEX e o OPEX, ao mesmo tempo em que manteve elevados níveis de QoS. A terceira questão, sobre o atendimento a demandas heterogêneas por serviços IoT, foi contemplada por meio da implementação de um modelo de NS que considera parâmetros específicos de cada *slice*, e.g., SF, TP e requisitos de latência. A avaliação indicou que a abordagem garantiu o isolamento lógico entre fatias e a priorização adequada dos serviços críticos.

As simulações realizadas no ambiente ns-3, com cenários realistas e integração de módulos personalizados para modelagem de redes não-3GPP, confirmaram a superioridade da abordagem proposta em relação às soluções tradicionais. As soluções baseadas em DRL evidenciaram robustez, escalabilidade e capacidade de generalização, enquanto a formulação MILP forneceu pontos de partida otimizados que aceleraram a convergência dos agentes e evitaram soluções subótimas. A combinação dessas abordagens representa uma contribuição inédita e relevante para o estado da arte, conferindo à tese um caráter distintamente inovador.

Além dos ganhos quantitativos observados nas métricas de desempenho, a pesquisa introduz um paradigma metodológico replicável para a modelagem e solução de problemas complexos em redes heterogêneas e dinâmicas. A arquitetura proposta, com hierarquia de decisão e integração entre NS, VANTs, DRL e otimização exata, pode ser estendida a diferentes domínios, como cidades inteligentes, redes emergenciais, computação ubíqua e sistemas distribuídos autônomos.

Conclui-se, portanto, que esta tese contribui de forma expressiva para o avanço científico e tecnológico da área, oferecendo uma solução técnica robusta, conceitualmente inovadora e experimentalmente validada para a alocação dinâmica de recursos em redes IoT não-3GPP com mobilidade e fatiamento. Os resultados confirmam a viabilidade e a superioridade da abordagem proposta, fundamentando sua aplicação prática e fomentando novas investigações na interface entre redes móveis e inteligência artificial.

6.1 Resultados Obtidos

A elaboração desta tese resultou em algumas produções acadêmicas e bibliográficas, refletindo o desenvolvimento e a disseminação do conhecimento gerado ao longo da pesquisa. Estas produções incluem publicações em periódicos científicos, apresenta-

ções em congressos e códigos em repositórios públicos. A seguir, são apresentadas as publicações e repositórios que foram desenvolvidos ao longo do trabalho.

Tabela 6.1: *Publicações no tema da tese.[Criada pelo Autor.]*

Nº	Produção	Tipo	Documento
1	R. S. Silva, W. Pires, S. L. Correa, A. Oliveira and K. V. Cardoso., (2023). Dynamic resource allocation in non-3GPP IoT networks involving UAVs. IEEE VTC2023-Spring, Florence, Italy. DOI:10.1109/VTC2023-Spring57618.2023.10199941	Conferência	https://ieeexplore.ieee.org/document/10199941
2	Pires-Jr, W. et al. (2024). QoS-aware Optimal Deployment of LoRa gateways in UAV-enabled LoRaWANs. SBRC 2024. DOI: 10.5753/sbrc.2024.1443	Conferência	https://sol.sbc.org.br/index.php/sbrc/article/view/29820
3	R. S. Silva et al. (2024). Soluções baseadas em aprendizado por reforço profundo para implantar VANTs como gateways LoRaWAN. Anais do XLII SBRT, v. 1. SBRT 2024. DOI: 10.14209/sbrc.2024.1571036460	Conferência	https://biblioteca.sbrc.org.br/articles/4668
4	R. S. Silva, A. C. Oliveira, K. V. Cardoso. (2025). Deep Reinforcement Learning-Driven UAV Orchestration for Dynamic Resource Allocation in Non-3GPP IoT Network slices. <i>IEEE Access</i> .	Periódico	Em processo de submissão

A Tabela 6.1 apresenta as publicações diretamente relacionadas ao tema da tese. Esses trabalhos enfocam aspectos fundamentais da pesquisa, como a alocação dinâmica de VANTs e otimização de redes IoT não-3GPP, sendo publicados em conferências e periódicos reconhecidos na área.

Além das publicações diretamente ligadas ao tema central da tese, também foram desenvolvidos trabalhos que, embora não sejam o foco principal da pesquisa, estão alinhados com suas direções e metodologias. A Tabela 6.2 lista essas publicações, que abordam tópicos como aprendizado federado em redes IoT, computação em borda e arquiteturas emergentes para redes 6G.

Além da produção acadêmica tradicional, um dos principais resultados desta pesquisa foi a implementação de ferramentas computacionais, disponibilizadas como código aberto para a comunidade acadêmica e profissional. Esses repositórios foram essenciais para a condução dos experimentos e possibilitam a reprodução e expansão dos resultados obtidos. A Tabela 6.3 apresenta os principais repositórios de código desenvolvidos ao longo do trabalho.

Estas contribuições destacam o impacto acadêmico e prático do trabalho, promovendo o conhecimento e sua aplicação em pesquisas futuras e projetos inovadores. Além disso, a disponibilização de código aberto fortalece a transparência científica e a reprodutibilidade dos experimentos, permitindo que a comunidade explore e amplie as soluções propostas nesta tese.

Tabela 6.2: *Publicações alinhadas ao tema da tese. [Criada pelo Autor.]*

Nº	Produção	Tipo	Resultado
1	Oliveira, R., S. e Silva, R. , Freitas, L., e Oliveira-Jr, A., (2024). Aprendizado Federado em Redes IoT sem Fio: Novo Algoritmo para a Seleção de Dispositivos e Alocação dos Recursos de Comunicação. In Anais do XLII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, (pp. 99-112). Porto Alegre: SBC. doi:10.5753/sbrc.2024.1267	Conferência	https://sol.sbc.org.br/index.php/sbrc/article/view/29786/29589
2	Xavier, R., Silva, R. S. , Ribeiro, M., Moreira, W., Freitas, L., e Oliveira-Jr, A. (2024). Integrating Multi-Access Edge Computing (MEC) into Open 5G Core. <i>Telecom</i> , 5(2), 433-450. https://doi.org/10.3390/telecom5020022	Periódico	https://www.mdpi.com/2673-4001/5/2/22/pdf?version=1717409425
3	Oliveira, R., S. e Silva, R. , Freitas, L., e Oliveira-Jr, A., (2024). Deep Q-Network para a Alocação dos Recursos de Comunicação do Aprendizado Federado em Redes IoT sem Fio. In: XLII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, Belém, PA, Brasil. Anais do XLII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2024. v. 1.	Conferência	https://biblioteca.sbrc.org.br/articles/4593
4	Pivoto D. G. S., Rezende T. T., Facina M. S. P., Silva, R. S. , Moreira, R., Silva F. O., Cardoso, K. V., Correa, S. L., Araújo, A. V. D., Scalco H. N., Tejerina, Gustavo, R. L., e Alberti, A.M., (2023). A Detailed Relevance Analysis of Enabling Technologies for 6G Architectures, in <i>IEEE Access</i> , vol. 11, pp. 89644-89684, doi: 10.1109/ACCESS.2023.3301811.	Periódico	https://ieeexplore.ieee.org/document/10207841
5	Eduardo, G., Luiz, A., S. e Silva, R. , Moreira, W., Freitas, L., e Oliveira-Jr, A. (2023). Fatiamento de rede utilizando MEC para integrar V2X por meio do acesso Non-3GPP à rede 6G (5G/B5G). In Anais do III Workshop de Redes 6G, (pp. 31-36). Porto Alegre: SBC. doi:10.5753/w6g.2023.752	Conferência	https://sol.sbc.org.br/index.php/w6g/article/view/24665/24486
6	Oliveira, R., S. e Silva, R. , Freitas, L., e Oliveira-Jr, A., (2023). Aprendizado Federado e Deep Q-Network habilitando VANTs como Infraestrutura em Redes 6G. In: Workshop de Redes 6G, Brasil. Anais do III Workshop de Redes 6G (W6G 2023). Porto Alegre, RS: SBC OpenLib, 2023. v. 1. p. 1-6.	Conferência	https://sol.sbc.org.br/index.php/w6g/article/view/24660/24481
7	P. Júnior, C., Silva, R. S. , Wanberg, S., Cardoso, K., e Oliveira-Jr, A. (2021). Implementação de uma rede IoT LoRaWAN para integração com o núcleo 5G. In Anais da IX Escola Regional de Informática de Goiás, (pp. 185-194). Porto Alegre: SBC. doi:10.5753/erigo.2021.18443	Conferência	https://sol.sbc.org.br/index.php/erigo/article/view/18443/18276
8	Lima, H., Silva, R. S. , Both, C., Oliveira-Jr, A., Cardoso, K., e Corrêa, S. (2021). Orquestração Inteligente de Network Slicing: Revisão da Literatura e Prospecção para Redes 6G. In Anais do I Workshop de Redes 6G, (pp. 19-24). Porto Alegre: SBC. doi:10.5753/w6g.2021.17230	Conferência	https://sol.sbc.org.br/index.php/w6g/article/view/17230/17068
9	Silva, J. P. L. ; Nery, S. W. L. ; Silva, R. S. ; Oliveira-Jr, A. C. ; Cardoso, K. V. ; Both, C. B. . (2020). Entendendo o núcleo 5G na prática, através de uma implementação de código aberto. In: Edmar Candeia Gurjão. (Org.). Entendendo o núcleo 5G na prática através de uma implementação de código aberto. 1ªed. Campina Grande, PB: Editora IFPB, v. 1, p. 10-41.	Cap. Livro	http://editora.ifpb.edu.br/ifpb/catalog/view/401/209/1168-2

Tabela 6.3: *Software e Repositórios de Código. [Criada pelo Autor.]*

Nº	Produção	Tipo	Resultado
1	Silva, R. S. (2024). <i>A2C-LoVQI</i>	Repositório	https://github.com/LABORA-INF-UFG/A2C-LoVQI
2	Silva, R. S. (2024). <i>NYUSIM in NS-3</i> .	Repositório	https://github.com/LABORA-INF-UFG/NYUSIM_in_ns3
3	Silva, R. S. (2024). <i>NS3-Gym</i> .	Repositório	https://github.com/LABORA-INF-UFG/ns3-gym
4	Silva, R. S. (2023). <i>Non-3GPP IoT Simulations</i> .	Repositório	https://github.com/LABORA-INF-UFG/non3GPP_IoT_simulations
5	Silva, R. S. (2022). <i>Sliced NS-3 LoRaWAN Module</i> .	Repositório	https://github.com/LABORA-INF-UFG/sliced-lorawan
6	Silva, J.P.L, Nery, S.W.S, e Silva, R. S. (2020) <i>SBrT2020 Minicurso</i>	Repositório	https://github.com/LABORA-INF-UFG/SBrT2020-Minicurso1

6.2 Trabalhos Futuros

O presente trabalho apresentou uma abordagem otimizada para a alocação dinâmica de recursos em fatias de redes **LoRaWAN** envolvendo **VANTs**. No entanto, diversas possibilidades de aprimoramento e expansão podem ser exploradas em investigações futuras.

Extensão do Modelo de Otimização

Embora o modelo proposto tenha demonstrado eficiência na alocação de recursos, sua formulação pode ser expandida para considerar fatores adicionais, tais como:

- Inclusão de restrições relacionadas ao consumo energético dos **LoRa-EDs** e dos **VANTs**, visando maior eficiência energética nas decisões de posicionamento e associação;
- Avaliação de modelos alternativos de propagação, com destaque para a incorporação de modelos de perdas com *shadowing*, que representam variações lentas na potência do sinal causadas por obstruções ambientais, sendo determinantes em cenários urbanos densos e aplicações que envolvem enlaces mais sensíveis às flutuações do canal;
- Estudo da viabilidade de incorporar o *shadowing* diretamente na formulação do problema de otimização, analisando o impacto dessa complexidade adicional sobre o desempenho computacional do solucionador exato (**MILP**).

Extensão das Técnicas de Inteligência Artificial

Uma das limitações dos métodos tradicionais de aprendizado de máquina aplicados à alocação dinâmica de recursos em redes **IoT** e **LoRaWAN** é a necessidade de cen-

tralização dos dados para o treinamento dos modelos. Essa abordagem pode ser inviável devido a restrições de largura de banda, consumo energético dos dispositivos e preocupações com privacidade. Para mitigar esses desafios, uma linha promissora de pesquisa futura é a adoção do *Federated Learning* (FL).

O aprendizado federado permite que modelos de ML sejam treinados de forma distribuída diretamente nos dispositivos IoT ou VANTs, sem a necessidade de transferir grandes volumes de dados para um servidor central. Para aplicar o aprendizado federado à alocação dinâmica de recursos, algumas direções podem ser exploradas:

- **Uso de VANTs como Agentes de Coordenação:** Os VANTs podem atuar como agregadores locais, coletando e processando modelos de aprendizado de dispositivos IoT em uma determinada região antes de consolidar as atualizações para o servidor central.
- **Implementação de Técnicas de Agregação de Modelos:** Estratégias como *Federated Averaging* (FedAVG) e *Personalized Federated Learning* (PFL) podem ser investigadas para otimizar a fusão dos modelos treinados localmente.
- **Adaptação ao Contexto da Rede:** Incorporar técnicas de aprendizado federado específicas para redes LoRaWAN, levando em consideração aspectos como disponibilidade intermitente dos dispositivos e restrições de energia.
- **Combinação com Aprendizado por Reforço:** Integrar aprendizado federado com DRL pode permitir que os dispositivos IoT e VANTs aprendam políticas ótimas de alocação de recursos de maneira descentralizada e eficiente.

A implementação dessas técnicas pode trazer ganhos significativos para redes IoT, permitindo uma alocação mais eficiente de recursos, redução da latência e maior autonomia dos dispositivos. Além disso, a abordagem descentralizada do aprendizado federado se alinha com o paradigma emergente da computação de borda, reduzindo a dependência de servidores centrais e possibilitando decisões inteligentes diretamente nos nós da rede.

Validação em ambiente real

Até o momento, a análise do desempenho foi conduzida por meio de simulações no ns-3. Como evolução do trabalho, sugere-se a realização de experimentos em um ambiente real, permitindo:

- Testes em plataformas de hardware LoRaWAN, avaliando a implementação prática da metodologia proposta;
- Uso de VANTs reais para validar a estratégia de posicionamento dinâmico e sua eficiência em diferentes cenários urbanos e rurais;

- Comparação com abordagens convencionais de alocação de recursos, visando quantificar os ganhos obtidos em condições reais de operação.

Referências Bibliográficas

- [3GPP 2016]3GPP. *Standardization of NB-IOT completed*. Jun 2016. Disponível em: <https://www.3gpp.org/news-events/3gpp-news/1785-nb_iot_complete>.
- [3GPP 2018]3GPP. *Telecommunication Management; Study on Management and Orchestration of Network Slicing for Next Generation Network*. Jan. 2018.
- [3GPP 2020]3GPP. *Study on channel model for frequencies from 0.5 to 100 GHz*. Sophia Antipolis CEDEX, France, 11 2020. Disponível em: <<https://www.etsi.org>>.
- [3GPP 2021]3GPP. *5G; Access to the 3GPP 5G Core Network (5GCN) via non-3GPP access networks*. [S.I.], 04 2021. Version: 16.7.0 - Release 16.
- [3GPP 2022]3GPP. *System architecture for the 5G System (5GS): 3GPP TS 23.501 version 17.4.0 Release 17*. Sophia Antipolis CEDEX, France, 05 2022. Disponível em: <https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/17.04.00_60>.
- [Ackerman e Koziol 2019]ACKERMAN, E.; KOZIOL, M. The blood is here: Zipline’s medical delivery drones are changing the game in Rwanda. *IEEE Spectrum*, v. 56, n. 5, p. 24–31, 2019.
- [Afolabi et al. 2018]AFOLABI, I. et al. Network Slicing and Softwarization: A Survey on Principles, Enabling Technologies, and Solutions. *IEEE Communications Surveys Tutorials*, v. 20, n. 3, p. 2429–2453, thirdquarter 2018. ISSN 1553-877X.
- [Al-Turjman et al. 2019]AL-TURJMAN, F. et al. Enhanced Deployment Strategy for the 5G Drone-BS Using Artificial Intelligence. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 7, p. 75999–76008, 2019. ISSN 21693536.
- [Almeida, Campos e Ricardo 2022]ALMEIDA, E. N.; CAMPOS, R.; RICARDO, M. Traffic-Aware UAV Placement using a Generalizable Deep Reinforcement Learning Methodology. *Proceedings - IEEE Symposium on Computers and Communications*, IEEE, v. 2022-June, p. 1–6, 2022. ISSN 15301346.

- [An e Wang 2024]AN, H.; WANG, L. Robust Topology Generation of Internet of Things Based on PPO Algorithm Using Discrete Action Space. *IEEE Transactions on Industrial Informatics*, v. 20, n. 4, p. 5406–5414, 2024.
- [Augustin et al. 2016]AUGUSTIN, A. et al. A Study of LoRa: Long Range, Low Power Networks for the Internet of Things. *Sensors*, v. 16, n. 9, 2016. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/16/9/1466>>.
- [Barakabitze et al. 2020]BARAKABITZE, A. A. et al. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Computer Networks*, v. 167, p. 106984, 2020. ISSN 1389-1286. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1389128619304773>>.
- [Bestuzheva et al. 2021]BESTUZHEVA, K. et al. *The SCIP Optimization Suite 8.0*. [S.l.], December 2021.
- [Boosters 2023]BOOSTERS, S. *Why your cell phone signal is weaker in a crowd*. June 2023. Accessed: 2024-12-29. Disponível em: <https://www.surecallboosters.ca/post/why-your-cell-phone-signal-is-weaker-in-a-crowd?srsltid=AfmBOorsZD6plkORESyTr2L3OU3FG5b5wWFzqsFc27d4XNe5HH9_dgRM>.
- [Bor et al. 2016]BOR, M. et al. Do LoRa Low-Power Wide-Area Networks Scale? In: *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. [S.l.: s.n.], 2016. p. 59–67.
- [Boucherie e Dijk 2017]BOUCHERIE, R. J.; DIJK, N. M. van. *Markov Decision Processes in Practice*. Cham, Switzerland: Springer, 2017. ISBN 978-3-319-55660-1.
- [Brasil 2017]BRASIL. Requisitos Gerais para Aeronaves Não Tripuladas de uso Civil. *Diário Oficial [da] República Federativa do Brasil*, Brasília, DF, 2017. Resolução nº 419, de 02.05.2017. [Em.02] em vigor em 1º de junho de 2022. Disponível em: <<https://www.anac.gov.br/assuntos/legislacao/legislacao-1/resolucoes/2017/resolucao-no-419-02-05-2017>>.
- [Bynum et al. 2021]BYNUM, M. L. et al. *Pyomo—optimization modeling in python*. Third. [S.l.]: Springer Science & Business Media, 2021.
- [Canva 2024]Canva. *Canva Dream Lab: AI-Generated Visuals*. 2024. Accessed: de 2024-08-01 até 2025-01-31. Disponível em: <<https://www.canva.com/features/dream>>.
- [Chartsias et al. 2017]CHARTSIAS, P. K. et al. Sdn/nfv-based end to end network slicing for 5g multi-tenant networks. In: *2017 European Conference on Networks and Communications (EuCNC)*. [S.l.: s.n.], 2017. p. 1–5.

- [Chen et al. 2021]CHEN, J. et al. Exploiting 5G and Blockchain for Medical Applications of Drones. *IEEE Network*, v. 35, n. 1, p. 30–36, 2021.
- [Das et al. 2021]DAS, D. N. et al. Synchronized Truck and Drone Routing in Package Delivery Logistics. *IEEE Transactions on Intelligent Transportation Systems*, v. 22, n. 9, p. 5772–5782, 2021.
- [Daud et al. 2018]DAUD, S. et al. Performance Evaluation of Low Cost LoRa Modules in IoT Applications. *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, v. 318, mar 2018.
- [Dawaliby, Bradai e Pousset 2019]DAWALIBY, S.; BRADAI, A.; POUSSET, Y. Adaptive dynamic network slicing in LoRa networks. *Future Generation Computer Systems*, North-Holland, v. 98, p. 697–707, sep 2019. ISSN 0167739X.
- [Dawaliby, Bradai e Pousset 2021]DAWALIBY, S.; BRADAI, A.; POUSSET, Y. Joint slice-based spreading factor and transmission power optimization in LoRa smart city networks. *Internet of Things (Netherlands)*, Elsevier, v. 14, p. 100–121, jun 2021. ISSN 25426605.
- [Dawaliby et al. 2018]DAWALIBY, S. et al. Dynamic network slicing for lorawan. In: *2018 14th International Conference on Network and Service Management (CNSM)*. [S.l.: s.n.], 2018. p. 134–142. ISSN 2165-963X.
- [Esteves et al. 2021]ESTEVEES, J. J. A. et al. DRL-based Slice Placement under Realistic Network Load Conditions. In: *International Conference on Network and Service Management (CNSM) DRL-based*. [S.l.]: Institute of Electrical and Electronics Engineers (IEEE), 2021. p. 524–526.
- [Europe 2019]EUROPE. The rules and procedures for the operation of unmanned aircraft. *Official Journal of the European Union*, European Union, 2019. COMMISSION IMPLEMENTING REGULATION (EU) 2019/947 of 24 May 2019. Disponível em: <<https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32019R0947>>.
- [Feriani e Hossain 2021]FERIANI, A.; HOSSAIN, E. Single and Multi-Agent Deep Reinforcement Learning for AI-Enabled Wireless Networks: A Tutorial. *IEEE Communications Surveys & Tutorials*, v. 23, n. 2, p. 1226–1252, 2021.
- [Gawłowicz e Zubow 2019]GAWŁOWICZ, P.; ZUBOW, A. ns-3 meets OpenAI Gym: The Playground for Machine Learning in Networking Research. In: *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. New York, NY, USA: Association for Computing Machinery, 2019. (MSWIM '19), p. 113–120. ISBN 9781450369046. Disponível em: <<https://doi.org/10.1145/3345768.3355908>>.

- [Gendia et al. 2024]GENDIA, A. et al. Energy-Efficient Trajectory Planning with Joint Device Selection and Power Splitting. *IEEE Transactions on Machine Learning in Communications and Networking*, IEEE, PP, p. 1, 2024.
- [George et al. 2019]GEORGE, S. et al. Towards Drone-Sourced Live Video Analytics for the Construction Industry. In: *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications*. New York, NY, USA: Association for Computing Machinery, 2019. (HotMobile '19), p. 3–8. ISBN 9781450362733.
- [GitHub 2024]GitHub. *GitHub Copilot: AI-Powered Coding Assistant*. 2024. Accessed: de 2024-08-01 até 2025-01-31. Disponível em: <<https://github.com/features/copilot>>.
- [Google 2025]GOOGLE, L. *Protocol buffers - Documentation*. [S.l.], 2025. Disponível em: <<https://protobuf.dev/>>.
- [Guardian 2023]GUARDIAN, T. *Why phone signal is still so scarce at UK music festivals*. August 2023. Accessed: 2024-12-29. Disponível em: <<https://www.theguardian.com/music/2023/aug/25/why-phone-signal-is-still-so-scarce-at-uk-music-festivals>>.
- [Gurobi 2025]GUROBI, L. *Gurobi Optimizer Reference Manual*. [S.l.], 2025. <https://www.gurobi.com>.
- [Hamdani, Fergani e Ramdani 2021]HAMDANI, S. E.; FERGANI, L.; RAMDANI, M. A Markov Decision Process Model for a Reinforcement Learning-Based Autonomous Pedestrian Crossing Protocol. In: *2021 International Conference on Advanced Communication Technologies and Networking (CommNet)*. [S.l.]: IEEE, 2021. p. 1–8.
- [He et al. 2015]HE, K. et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2015. p. 1026–1034.
- [He et al. 2018]HE, X. et al. Towards 3D Deployment of UAV Base Stations in Uneven Terrain. In: *2018 27th International Conference on Computer Communication and Networks (ICCCN)*. [S.l.: s.n.], 2018. p. 1–9.
- [Horizonte 2022]HORIZONTE, C. M. de B. *Falha de internet em grandes eventos demanda esforço conjunto para solução*. July 2022. Accessed: 2024-12-29. Disponível em: <<https://bit.ly/cmbh-comunicacao-falha-internet-2022>>.
- [Hu e Yue 2008]HU, Q.; YUE, W. *Markov Decision Processes With Their Applications*. London, UK: Springer, 2008. ISBN 978-1-84800-316-3.

- [InfoMoney 2024]INFOMONEY. *Tragédia no Rio Grande do Sul: chuvas afetam telecomunicações e dificultam resgates*. May 2024. Accessed: 2024-12-29. Disponível em: <<https://bit.ly/infomoney-tragedia-rio-grande-do-sul-2024>>.
- [ITU-T 2012]ITU-T. *Framework of network virtualization for future networks*. [S.l.], 2012. Disponível em: <<https://www.itu.int/rec>>.
- [ITU-T 2018]ITU-T. *IMT-2020 network management and orchestration requirements*. [S.l.], 2018. Disponível em: <<https://www.itu.int/rec>>.
- [Jan, Hassan e Jung 2019]JAN, M. A.; HASSAN, S. A.; JUNG, H. Qos-based performance analysis of mmwave uav-assisted 5g hybrid heterogeneous network. In: *2019 IEEE Global Communications Conference (GLOBECOM)*. [S.l.: s.n.], 2019. p. 1–6. ISSN 2576-6813.
- [Jang et al. 2019]JANG, B. et al. Q-learning algorithms: A comprehensive classification and applications. *IEEE Access*, v. 7, p. 133653–133667, 2019.
- [JetBrains 2024]JETBRAINS. *Serviço JetBrains AI Assistant*. 2024. <https://www.jetbrains.com/pt-br/ai/>. [Acesso em: 25-12-2024].
- [Jez et al. 2021]JEZ, L. et al. CSMA Adaptativo para Redes LoRa. In: *Anais do XXVI Workshop de Gerência e Operação de Redes e Serviços*. Porto Alegre, RS, Brasil: SBC, 2021. p. 68–81. ISSN 2595-2722. Disponível em: <<https://sol.sbc.org.br/index.php/wgrs/article/view/17186>>.
- [Khan et al. 2020]KHAN, S. K. et al. UAV-aided 5G Network in Suburban, Urban, Dense Urban, and High-rise Urban Environments. In: *2020 IEEE 19th International Symposium on Network Computing and Applications (NCA)*. [S.l.: s.n.], 2020. p. 1–4.
- [Kirubakaran et al. 2023]KIRUBAKARAN, B. et al. Optimized UAV-Based Connectivity Solutions for Urban IoT Networks. *International Congress on Ultra Modern Telecommunications and Control Systems and Workshops*, IEEE Computer Society, p. 1–6, 2023. ISSN 2157023X.
- [Klaine et al. 2018]KLAINÉ, P. V. et al. Distributed drone base station positioning for emergency cellular networks using reinforcement learning. *Cognitive Computation*, v. 10, n. 5, p. 790–804, Oct 2018. ISSN 1866-9964. Disponível em: <<https://doi.org/10.1007/s12559-018-9559-8>>.
- [Koushik, Hu e Kumar 2019]KOUSHIK, A. M.; HU, F.; KUMAR, S. Deep q-learning-based node positioning for throughput-optimal communications in dynamic UAV swarm network.

- IEEE Transactions on Cognitive Communications and Networking*, Institute of Electrical and Electronics Engineers Inc., v. 5, n. 3, p. 554–566, 2019. ISSN 23327731.
- [Kufakunesu, Hancke e Abu-Mahfouz 2020]KUFAKUNESU, R.; HANCKE, G. P.; ABU-MAHFOUZ, A. M. A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges. *Sensors*, v. 20, n. 18, 2020. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/20/18/5044>>.
- [Lee et al. 2014]LEE, W.-Y. et al. Spatial modeling of the traffic density in cellular networks. *Wireless Networks*, Springer, v. 20, n. 6, p. 1165–1176, 2014.
- [Li, Raza e Khan 2018]LI, S.; RAZA, U.; KHAN, A. How Agile is the Adaptive Data Rate Mechanism of LoRaWAN? In: *2018 IEEE Global Communications Conference (GLOBECOM)*. [S.l.: s.n.], 2018. p. 206–212. ISSN 2576-6813.
- [Lien e Deng 2022]LIEN, S.-Y.; DENG, D.-J. Autonomous Non-Terrestrial Base Station Deployment for Non-Terrestrial Networks: A Reinforcement Learning Approach. *IEEE Transactions on Vehicular Technology*, v. 71, n. 10, p. 10894–10909, 2022.
- [Lin et al. 2024]LIN, Y. M. et al. Joint Caching, Communication and Trajectory Optimization in Air-Ground Integrated Wireless Networks with Multiple UAVs and Multiple BSs. *IEEE Access*, IEEE, v. 12, p. 60095–60111, 2024. ISSN 21693536.
- [Liu, Ding e Liu 2021]LIU, Y.; DING, J.; LIU, X. Resource Allocation Method for Network Slicing Using Constrained Reinforcement Learning. In: *2021 IFIP Networking Conference (IFIP Networking)*. [S.l.: s.n.], 2021. p. 1–3.
- [Lobianco et al. 2021]Lobianco, J. P. et al. Entendendo o núcleo 5G na prática, através de uma implementação de código aberto. In: *SBRT 2020: Livro de Minicursos*. Florianópolis-SC: Instituto Federal de Ensino, Ciência e Tecnologia da Paraíba – IFPB, 2021. cap. 1. ISBN 978-65-87572-23-9.
- [Lodhi, Wang e Farhad 2022]LODHI, M. A.; WANG, L.; FARHAD, A. ND-ADR: Nondestructive adaptive data rate for LoRaWAN Internet of Things. *International Journal of Communication Systems*, v. 35, n. 9, p. e5136, 2022. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.5136>>.
- [LoRa Alliance 2015]LoRa Alliance. *A technical overview of LoRa and LoRaWAN: What is it?* San Ramon, CA, 2015. v. 1, 1–20 p. Disponível em: <https://lora-alliance.org/resource_hub/what-is-lorawan/>.
- [LoRa Alliance 2020]LoRa Alliance. *LoRaWAN Specification*. Fremont, CA, 10 2020. Disponível em: <<https://resources.lora-alliance.org/technical-specifications>>.

- [LoRa Alliance 2021]LoRa Alliance. *LoRaWAN® Regional Parameters*. San Ramon, CA, 05 2021. Disponível em: <https://lora-alliance.org/resource_hub/rp2-1-0-3-lorawan-regional-parameters/>.
- [Magrin, Centenaro e Vangelista 2017]MAGRIN, D.; CENTENARO, M.; VANGELISTA, L. Performance evaluation of LoRa networks in a smart city scenario. In: *Proc. IEEE Int. Conf. Comm. (ICC)*. [S.l.: s.n.], 2017. p. 1–7.
- [Mahadevan 2009]MAHADEVAN, S. Learning Representation and Control in Markov Decision Processes: New Frontiers. *Foundations and Trends® in Machine Learning*, v. 1, n. 4, p. 403–565, 2009. ISSN 1935-8237.
- [Marchese, Moheddine e Patrone 2019]MARCHESE, M.; MOHEDDINE, A.; PATRONE, F. IoT and UAV Integration in 5G Hybrid Terrestrial-Satellite Networks. *Sensors 2019, Vol. 19, Page 3704*, Multidisciplinary Digital Publishing Institute, v. 19, n. 17, p. 3704, aug 2019. ISSN 1424-8220. Disponível em: <<https://www.mdpi.com/1424-8220/19/17/3704/htmhttps://www.mdpi.com/1424-8220/19/17/3704>>.
- [Marchese, Moheddine e Patrone 2019]MARCHESE, M.; MOHEDDINE, A.; PATRONE, F. Towards Increasing the LoRa Network Coverage: A Flying Gateway. *2019 International Symposium on Advanced Electrical and Communication Technologies, ISAECT 2019*, Institute of Electrical and Electronics Engineers Inc., nov 2019.
- [Marchese, Moheddine e Patrone 2020]MARCHESE, M.; MOHEDDINE, A.; PATRONE, F. UAV and Satellite Employment for the Internet of Things Use Case. In: *2020 IEEE Aerospace Conference*. [S.l.: s.n.], 2020. p. 1–8. ISSN 1095-323X.
- [Mardi et al. 2022]MARDI, F. Z. et al. An Efficient Allocation System for Centralized Network Slicing in LoRaWAN. In: *Proc. Int. Wirel. Commun. Mob. Comput. Conf. (IWCMC)*. [S.l.: s.n.], 2022. p. 806–811.
- [Messaoud, Bradai e Lagrange 2021]MESSAOUD, A.; BRADAI, A.; LAGRANGE, X. Deep Federated Q-Learning-Based Network Slicing for Industrial IoT. *IEEE Internet of Things Journal*, IEEE, v. 8, n. 5, p. 3563–3575, 2021.
- [Messaoud, Bradai e Lagrange 2021]MESSAOUD, A.; BRADAI, A.; LAGRANGE, X. Slicing Optimization Based on Machine Learning Tool for Industrial IoT 4.0. *Journal of Network and Computer Applications*, Elsevier, v. 175, p. 102890, 2021.
- [Messaoud et al. 2021]MESSAOUD, A. et al. In-Depth Performance Evaluation of Network Slicing Strategies in Large Scale Industry 4.0. *IEEE Transactions on Industrial Informatics*, IEEE, v. 17, n. 5, p. 3500–3512, 2021.

- [Moon et al. 2023]MOON, J. et al. Implementation of mmWave long-range backhaul for UAV-BS. *ETRI Journal*, Wiley Online Library, v. 45, n. 5, p. 781–794, 2023.
- [Mozaffari et al. 2016]MOZAFFARI, M. et al. Optimal transport theory for power-efficient deployment of unmanned aerial vehicles. In: *2016 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2016. p. 1–6. ISSN 1938-1883.
- [Mozny et al. 2020]MOZNY, R. et al. Unifying Multi-Radio Communication Technologies to Enable mMTC Applications in B5G Networks. In: *2020 2nd 6G Wireless Summit (6G SUMMIT)*. [S.l.: s.n.], 2020. p. 1–5.
- [NGMN Alliance 2016]NGMN Alliance. *Description of Network Slicing Concept*. 2016.
- [OpenAI 2024]OpenAI. *ChatGPT: Conversational AI Model*. 2024. Accessed: de 2024-08-01 até 2025-01-31. Disponível em: <<https://openai.com/chatgpt>>.
- [Parvaresh e Kantarci 2023]PARVARESH, N.; KANTARCI, B. A Continuous Actor-Critic Deep Q-Learning-Enabled Deployment of UAV Base Stations: Toward 6G Small Cells in the Skies of Smart Cities. *IEEE Open Journal of the Communications Society*, Institute of Electrical and Electronics Engineers Inc., v. 4, p. 700–712, 2023. ISSN 2644125X.
- [Pellegrini e Wainer 2007]PELLEGRINI, J.; WAINER, J. Processos de Decisão de Markov: um tutorial. *Revista de Informática Teórica e Aplicada; Vol. 14, No 2 (2007); 133-179*, v. 14, 12 2007.
- [Pires-Jr et al. 2024]PIRES-JR, W. et al. QoS-aware Optimal Deployment of LoRa Gateways in UAV-enabled LoRaWANs. In: *Anais do XLII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*. Porto Alegre, RS, Brasil: SBC, 2024. p. 574–587. ISSN 2177-9384. Disponível em: <<https://sol.sbc.org.br/index.php/sbrc/article/view/29820>>.
- [Popovski et al. 2018]POPOVSKI, P. et al. 5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View. *IEEE Access*, v. 6, p. 55765–55779, 2018.
- [PPIC 2023]PPIC, P. P. I. of C. *Crisis Communication Suffers During Natural Disasters*. January 2023. Accessed: 2024-12-29. Disponível em: <<https://www.ppic.org/blog/crisis-communication-suffers-during-natural-disasters/>>.
- [Ray e Nguyen 2020]RAY, P. P.; NGUYEN, K. A Review on Blockchain for Medical Delivery Drones in 5G-IoT Era: Progress and Challenges. In: *2020 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*. [S.l.: s.n.], 2020. p. 29–34.

- [Raza et al. 2018]RAZA, M. R. et al. Dynamic slicing approach for multi-tenant 5G transport networks. *Journal of Optical Communications and Networking*, Optica Publishing Group, v. 10, n. 1, p. A77–A90, 2018.
- [Rezazadeh 2024]REZAZADEH, A. A Mathematical Introduction to Deep Reinforcement Learning for 5G/6G Applications. *IEEE Communications Surveys and Tutorials*, IEEE, v. 26, n. 1, p. 1–35, 2024.
- [Ribeiro et al. 2021]RIBEIRO, E. et al. Optimization of Inspections of Underground Electricity Distribution Networks Using UAV. In: *2021 9th International Electrical Engineering Congress (iEECON)*. [S.l.: s.n.], 2021. p. 89–92.
- [Ruetten et al. 2020]RUETTEN, L. et al. Area-Optimized UAV Swarm Network for Search and Rescue Operations. In: *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. [S.l.: s.n.], 2020. p. 0613–0618.
- [Schedl, Kurmi e Bimber 2021]SCHEDL, D. C.; KURMI, I.; BIMBER, O. An autonomous drone for search and rescue in forests using airborne optical sectioning. *Science Robotics*, v. 6, n. 55, p. eabg1188, 2021. Disponível em: <<https://www.science.org/doi/abs/10.1126/scirobotics.abg1188>>.
- [Schulman et al. 2017]SCHULMAN, J. et al. Proximal Policy Optimization Algorithms. *arXiv preprint arXiv:1707.06347*, 2017. Disponível em: <<https://arxiv.org/abs/1707.06347>>.
- [Semtech 2015]Semtech. *LoRa™ Modulation Basics*. [S.l.], may 2015. 1–26 p. Disponível em: <<https://semtech.my.salesforce.com/sfc/>>.
- [Semtech 2019]Semtech. *LoRa and LoRaWAN: Technical overview | DEVELOPER PORTAL*. Camarillo, CA, 2019. 26 p. Disponível em: <<https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/>>.
- [Semtech 2019]Semtech. *Understanding the LoRa Adaptive Data Rate*. Camarillo, CA, 2019. 15 p. Disponível em: <<https://lora-developers.semtech.com/documentation/tech-papers-and-guides/understanding-adr/>>.
- [Shakhatreh e Khreishah 2018]SHAKHATREH, H.; KHREISHAH, A. Optimal Placement of a UAV to Maximize the Lifetime of Wireless Devices. In: *2018 14th International Wireless Communications Mobile Computing Conference (IWCMC)*. [S.l.: s.n.], 2018. p. 1225–1230. ISSN 2376-6506.
- [Sharma, Haque e Blaabjerg 2021]SHARMA, H.; HAQUE, A.; BLAABJERG, F. Machine Learning in Wireless Sensor Networks for Smart Cities: A Survey. *Electronics*, v. 10, n. 9, 2021. ISSN 2079-9292. Disponível em: <<https://www.mdpi.com/2079-9292/10/9/1012>>.

- [SigFox 2021]SigFox. *SIGFOX*. Jun 2021. Disponível em: <<https://www.sigfox.com/en>>.
- [Silva et al. 2024]SILVA, R. S. et al. Soluções baseadas em Aprendizado por reforço profundo para implantar VANTs como gateways LoRaWAN com foco na qualidade de serviço IOT. In: *Anais do XLII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT)*. [S.l.: s.n.], 2024. p. 1–6.
- [Silva et al. 2023]SILVA, R. S. et al. Dynamic resources allocation in non-3GPP IoT networks involving UAVs. In: *Proc. IEEE Veh. Technol. Conf. (VTC)*. [S.l.: s.n.], 2023. p. 1–5.
- [Song et al. 2024]SONG, F. et al. Energy-Efficient Trajectory Optimization with Wireless Charging in UAV-Assisted MEC Based on Multi-Objective Reinforcement Learning. *IEEE Transactions on Mobile Computing*, IEEE, PP, p. 1–18, 2024. ISSN 15580660.
- [Su 2022]SU, Y. UAV-based 5G Air-to-Ground Integrated Wireless Network Coverage. In: *2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCoM/CyberSci-Tech)*. [S.l.: s.n.], 2022. p. 1–4.
- [Sundaram, Du e Zhao 2020]SUNDARAM, J. P. S.; DU, W.; ZHAO, Z. A Survey on LoRa Networking: Research Problems, Current Solutions, and Open Issues. *IEEE Communications Surveys and Tutorials*, v. 22, n. 1, p. 371–388, Firstquarter 2020. ISSN 1553-877X.
- [Sundaram, Du e Zhao 2020]SUNDARAM, J. P. S.; DU, W.; ZHAO, Z. A Survey on LoRa Networking: Research Problems, Current Solutions, and Open Issues. *IEEE Communications Surveys and Tutorials*, v. 22, n. 1, p. 371–388, Firstquarter 2020. ISSN 1553-877X.
- [Sutton e Barto 2018]SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction*. 2nd. ed. Cambridge, MA: MIT Press, 2018. ISBN 978-0262039246. Disponível em: <<http://incompleteideas.net/book/the-book-2nd.html>>.
- [Tellache et al. 2022]TELLACHE, A. et al. Deep Reinforcement Learning based Resource Allocation in Dense Sliced LoRaWAN Networks. In: *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*. [S.l.: s.n.], 2022. p. 1–6.
- [Toscano et al. 2019]TOSCANO, M. et al. Machine Learning Aided Network Slicing. In: *2019 21st International Conference on Transparent Optical Networks (ICTON)*. [S.l.: s.n.], 2019. p. 1–4. ISSN 2161-2064.

- [Troudi et al. 2017]TROUDI, A. et al. Post-production analysis approach for drone delivery fleet. In: *2017 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*. [S.l.: s.n.], 2017. p. 150–155.
- [Wang et al. 2023]WANG, Y. et al. Caching Placement Optimization in UAV-Assisted Cellular Networks: A Deep Reinforcement Learning-Based Framework. *IEEE Wireless Communications Letters*, v. 12, n. 8, p. 1359–1363, 2023.
- [Wang et al. 2020]WANG, Z. et al. Utility Optimization for Resource Allocation in Edge Network Slicing Using DRL. *2020 IEEE Global Communications Conference, GLOBECOM 2020 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 12 2020.
- [Wijethilaka e Liyanage 2021]WIJETHILAKA, S.; LIYANAGE, M. Survey on Network Slicing for Internet of Things Realization in 5G Networks. *IEEE Communications Surveys and Tutorials*, v. 23, n. 2, p. 957–994, Secondquarter 2021. ISSN 1553-877X.
- [Xu et al. 2023]XU, X. et al. Joint Multi-UAV Deployment and Resource Allocation Based on Personalized Federated Deep Reinforcement Learning. *IEEE International Conference on Communications*, Institute of Electrical and Electronics Engineers Inc., v. 2023-May, p. 5677–5682, 2023. ISSN 15503607.
- [Zai e Brown 2020]ZAI, A.; BROWN, B. *Deep Reinforcement Learning in Action*. [S.l.]: Manning Publications, 2020. (AI in Action Series).
- [Zeng et al. 2020]ZENG, Y. et al. *UAV Communications for 5G and Beyond*. Wiley, 2020. (IEEE Press). ISBN 9781119575696. Disponível em: <<https://books.google.com.br/books?id=PQaivgEACAAJ>>.
- [Zhang 2019]ZHANG, S. An Overview of Network Slicing for 5G. *IEEE Wireless Communications*, v. 26, n. 3, p. 111–117, June 2019. ISSN 1558-0687.
- [Zhong et al. 2021]ZHONG, X. et al. QoS-Compliant 3-D Deployment Optimization Strategy for UAV Base Stations. *IEEE Systems Journal*, Institute of Electrical and Electronics Engineers Inc., v. 15, n. 2, p. 1795–1803, jun 2021. ISSN 19379234.
- [Zhou et al. 2020]ZHOU, F. et al. Automatic network slicing for iot in smart city. *IEEE Wireless Communications*, v. 27, n. 6, p. 108–115, December 2020. ISSN 1558-0687.
- [Zhou et al. 2023]ZHOU, G. et al. Multi-objective Optimization of Space-Air-Ground Integrated Network Slicing Relying on a Pair of Central and Distributed Learning Algorithms. *ResearchGate*, n. September, 2023.

Uso de Ferramentas de Inteligência Artificial

No desenvolvimento desta tese, ferramentas de IA como *Canva Dream Lab* [Canva 2024], *ChatGPT* [OpenAI 2024], *GitHub Copilot* [GitHub 2024] e *Jetbrains AI Assistant* [JetBrains 2024], foram utilizadas como suporte em tarefas operacionais e repetitivas, sem comprometer a integridade da pesquisa ou a autoria intelectual do trabalho. O *Canva Dream Lab* foi empregado para criar representações visuais complementares, facilitando a ilustração de conceitos complexos. O *ChatGPT* auxiliou na geração de código \LaTeX para a inserção de figuras, tabelas e diagramas, contribuindo para uma formatação eficiente e precisa dos elementos gráficos e estruturais do documento, bem como para revisão ortográfica e gramatical. Já o *GitHub Copilot* e o *JetBrains AI Assistant* foram utilizados para automação de trechos de códigos técnicos empregados nas simulações, bem como para automatizar ajustes em *shell scripts* complexos e como auxílio na correção de erros, otimizando o desenvolvimento dos diversos códigos. É importante destacar que o uso dessas ferramentas esteve restrito ao apoio técnico e operacional, sendo todo o conteúdo analítico, conceitual e crítico da tese fruto exclusivo do trabalho intelectual do autor. Desta forma, o uso das ferramentas garantiu maior eficiência nas etapas acessórias da elaboração, sem impactar a originalidade ou o rigor científico da pesquisa.

A.1 Canva Dream-Lab.

O *Canva Dream Lab* é uma plataforma que utiliza IA para criar gráficos e imagens geradas automaticamente a partir de um comando de descrição apresentado pelo usuário [Canva 2024]. Durante o desenvolvimento desta tese, a ferramenta foi empregada para produzir representações visuais que complementam os conceitos discutidos, como diagramas explicativos e cenários ilustrativos. Essas imagens foram cuidadosamente revisadas para garantir que refletissem com precisão os temas abordados, mantendo a integridade e a coerência com o conteúdo científico.

Figura Representativa para Evento Musical

Comando: “Crie um cenário vibrante de um grande evento musical em um estádio lotado. A imagem deve capturar uma perspectiva aérea ampla e distante, incluindo o estacionamento repleto de carros, pessoas chegando e se organizando em filas nos arredores do estádio. Muitas pessoas dentro do estádio, próximas ao palco. Mostre barracas coloridas vendendo *souvenirs* e comidas, com destaque para o movimento intenso e animado. Inclua uma estação de metrô e pontos de ônibus próximos, com pessoas caminhando para o evento. Adicione ambulâncias e carros de polícia posicionados estrategicamente, garantindo a segurança, e torres de luz iluminando o ambiente. Dentro do estádio, mostre a plateia agitada e o palco grandioso com luzes e equipamentos, transmitindo a energia do espetáculo. Capture o entusiasmo e a magnitude do evento em um cenário noturno”.



Figura A.1: Resultados para o comando: “Evento musical”.

Resultado: A Figura A.1 apresenta os resultados originais da ferramenta. A imagem da direita foi escolhida e editada na própria ferramenta. Os ícones e detalhes relacionados à rede de comunicação 3GPP e não-3GPP, aos VANTs, às BSs, às conexões e aos dispositivos da IoT foram adicionados posteriormente com o uso da ferramenta de Desenhos do Google.

Figura Representativa para Evento Esportivo

Comando: “Crie um cenário vibrante de um evento esportivo de Fórmula 1 no Autódromo de Interlagos, São Paulo. A imagem deve capturar uma perspectiva aérea, ampla e distante que inclua a movimentação intensa ao redor do local: carros chegando ao autódromo e pessoas acampando e fazendo churrasco no gramado. Mostre a arquibancada lotada de torcedores animados, com bandeiras e adereços temáticos, enquanto os carros estão posicionados no *grid* de largada, prontos para a corrida. Inclua detalhes da área dos boxes, com técnicos e mecânicos trabalhando freneticamente nos últimos ajustes dos carros. Capture o dinamismo, a energia e a grandiosidade do evento em um dia ensolarado, com destaque para a famosa curva do “S do Senna” ao fundo”.

Resultado: A Figura A.2 apresenta os resultados originais da ferramenta. A imagem da direita foi escolhida e editada na própria ferramenta *Canva Dream Lab*. Os ícones e detalhes relacionados à rede de comunicação 3GPP e não-3GPP, aos VANTs, às



Figura A.2: Resultados para o comando: “Evento Esportivo”.

BSs, às conexões e aos dispositivos da IoT foram adicionados posteriormente com o uso da ferramenta de Desenhos do Google.

Figura Representativa para Deslizamento de encosta

Comando: “Crie um cenário de deslizamento de terra em uma encosta próxima a uma favela urbana. A imagem deve ter uma perspectiva ampla, mostrando o impacto do deslizamento sobre as moradias simples e vulneráveis. Inclua equipes de resgate, bombeiros e moradores locais trabalhando juntos para procurar sobreviventes e ajudar os desabrigados. Adicione escavadeiras e veículos de emergência auxiliando nos esforços de resgate, enquanto parte das construções permanece soterrada. Mostre detalhes como chuvas intensas ao fundo, árvores e vegetação caídas, e áreas alagadas que intensificam a cena de desastre. Capture a solidariedade humana em meio à tragédia, com pessoas oferecendo ajuda e conforto”.



Figura A.3: Resultados para o comando “Deslizamento de encosta”.

Resultado: A Figura A.3 mostra os resultados originais da ferramenta. A imagem da esquerda foi escolhida e editada na própria ferramenta *Canva Dream Lab*. Os ícones e detalhes relacionados à rede de comunicação 3GPP e não-3GPP, aos VANTs, às BSs, às conexões e aos dispositivos da IoT foram adicionados posteriormente com o uso da ferramenta de Desenhos do Google.

Figura Representativa para Incêndio na Zona Rural

Comando: “Crie um cenário de queimadas em uma zona rural com uma perspectiva ampla e distante. A imagem deve retratar o combate a um incêndio florestal em uma área de cerrado no Pantanal mato-grossense. Inclua uma lavoura de soja parcialmente consumida pelas chamas, com veículos especializados e bombeiros combatendo os

focos de incêndio. Mostre animais típicos do cerrado, como uma onça-pintada, tucanos, araras-azuis e cobras sucuris, fugindo das chamas ou sendo resgatados por voluntários e veterinários. No céu, um helicóptero ao longe auxilia no combate, transportando água ou materiais. Capture a dramaticidade do fogo e o esforço humano e animal em meio à paisagem característica da região”.



Figura A.4: Resultados para o comando “Incêndio na zona rural”.

Resultado: A Figura A.4 mostra os resultados originais da ferramenta. A segunda imagem, da esquerda para a direita, foi escolhida e editada na própria ferramenta *Canva Dream Lab*. Os ícones e detalhes relacionados à rede de comunicação 3GPP e não-3GPP, aos VANTs, às BSs, às conexões e aos dispositivos da IoT foram adicionados posteriormente com o uso da ferramenta de Desenhos do Google.

A.2 OpenAI ChatGPT (GPT-4o)

O *ChatGPT* é um modelo de linguagem baseado em IA, desenvolvido pela *OpenAI*, que utiliza redes neurais profundas para compreender e gerar texto de forma contextualizada em linguagem natural [OpenAI 2024]. No contexto desta tese, o *ChatGPT* foi utilizado na geração de código \LaTeX para a inserção de figuras, tabelas e diagramas, contribuindo para uma formatação eficiente e precisa dos elementos gráficos e estruturais do documento, bem como para revisão ortográfica e gramatical, garantindo a consistência e a padronização do documento. Embora sua aplicação tenha sido estritamente operacional, o uso do *ChatGPT* permitiu maior eficiência na formatação e estruturação do conteúdo técnico. Ressalta-se que todas as análises, interpretações e formulações conceituais apresentadas neste trabalho foram exclusivamente desenvolvidas pelo autor.

Código \LaTeX para Diagrama de Estados e Transições

Comando: Analise o texto a seguir, delimitado entre aspas após o identificador “TEXTO:”. Este conteúdo é correspondente a uma seção de referencial teórico no tema “Processos de decisão de Markov (MDP)”. Sugira uma figura para o diagrama de transição de estados que demonstre os conceitos fundamentais acerca dos MDPs e que esteja alinhado com o conteúdo fornecido. Adote uma formalidade matemática para essa

classe de diagramas. Crie um *script* \LaTeX para a figura que utilize o pacote *TikZ*. O *label* da figura será “fig:cap_II_state_transitions_MDP”, o *caption* será “Diagrama de estados e transições para um Processo de Decisão Markoviano (MDP)”. Adicione ao menos 5 estados e um número de transições suficientes para demonstrar os conceitos discutidos na seção.

Texto descritivo do diagrama: “A Figura \ref{fig:cap_II_state_transitions_MDP} representa as interações fundamentais em um **MDP**, destacando os estados \mathcal{S} , ações \mathcal{A} , probabilidades de transição \mathcal{T} e recompensas \mathcal{R} . Esse modelo é essencial para o aprendizado por reforço, no qual políticas são aprendidas para maximizar recompensas acumuladas”.

Texto da seção:

```

 $\text{\section}$ {Processos de Decisão de Markov}
 $\text{\label}$ {sec:markov}
Os Processos de Decisão de Markov ( $\text{\ac}$ {mdp}) são ...
... No contexto de redes  $\text{\ac}$ {IoT}, o  $\text{\ac}$ {mdp} ...
 $\text{\%}$  $\text{\%}$  O texto completo da seção foi submetido como PROMPT,
 $\text{\%}$  $\text{\%}$  porém foi suprimido neste apêndice''.
```

Resultado: A Listagem A.1 demonstra o resultado do GPT.

Listagem A.1: Código em *LaTeX TikZ* gerado pelo ChatGPT

```

 $\text{\begin}$ {figure}[ht]
   $\text{\centering}$ 
   $\text{\begin}$ {tikzpicture}[
    state/.style={circle, draw,
      minimum size=1.2cm,
      font= $\text{\sffamily}$ \small},
    action/.style={draw,  $\text{\text{-}}\{\text{\text{Latex}}[scale=1.2]\}$ },
      font= $\text{\sffamily}$ \scriptsize}]
  % States
  \node[state] (S1) at (0, 0) {\$s_1\$};
  \node[state] (S2) at (3, 2) {\$s_2\$};
  \node[state] (S3) at (6, 0) {\$s_3\$};
  \node[state] (S4) at (3, -2) {\$s_4\$};
  \node[state] (S5) at (8, -2) {\$s_5\$};
  % Transitions with actions and probabilities
  \draw[action] (S1) -- node[midway, above] {\$a_1, \
    mathcal{T}=0.8, \mathcal{R}=10\$} (S2);
  \draw[action] (S2) -- node[midway, above] {\$a_2, \
    mathcal{T}=0.6, \mathcal{R}=20\$} (S3);
```

```

\draw[action] (S3) -- node[midway, below] {$a_3, \
  \mathcal{T}=0.7, \mathcal{R}=15$} (S4);
\draw[action] (S4) -- node[midway, above] {$a_4, \
  \mathcal{T}=0.5, \mathcal{R}=5$} (S5);
% Feedback loops and internal transitions
\draw[action, dashed, loop right] (S3) to node[right] {$
  a_5, \mathcal{T}=0.3, \mathcal{R}=8$} (S3);
\draw[action] (S4) to[bend right] node[midway, left] {$a
  _6, \mathcal{T}=0.2, \mathcal{R}=-5$} (S1);
% Additional feedback to a previous state
\draw[action, dashed] (S5) to[bend left] node[midway,
  below right] {$a_7, \mathcal{T}=0.4, \mathcal{R}=12$}
  (S3);
\end{tikzpicture}
\caption{Diagrama de estados e transições para um Processo
  de Decisão Markoviano (MDP).}
\label{fig:cap_II_state_transitions_MDP}
\end{figure}

```

Estruturação de citações e referências no formato $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X BibTeX}$

Comando: “Formate a referência *BibiTeX* e a citação para a notícia em <https://www.infomoney.com.br/politica/tragedia-no-rio-grande-do-sul-chuvas-afetam-telecomunicacoes-e-dificultam-resgates/>”.

Resultado: A Listagem A.2 demonstra o resultado do GPT.

Listagem A.2: Resultado gerado pelo ChatGPT

```

Aqui está uma citação LaTeX para a notícia mencionada:
\cite{infomoney2024}
Aqui está a referência bibliográfica para o BibTeX:
@misc{infomoney2024,
  author      = {InfoMoney},
  title       = {Tragédia no Rio Grande do Sul: chuvas
    afetam telecomunicações e dificultam resgates},
  year        = {2024},
  url         = {https://www.infomoney.com.br/politica/
    tragedia-no-rio-grande-do-sul-chuvas-afetam-
    telecomunicacoes-e-dificultam-resgates/},
  note        = {Accessed: 2024-12-29}
}

```

A.3 Assistente de código JetBrains AI Assistant

O *JetBrains AI* é uma ferramenta de assistência baseada em inteligência artificial integrada ao ecossistema de desenvolvimento da *JetBrains*, oferecendo suporte contextual para programação, refatoração de código e otimização de fluxos de trabalho [JetBrains 2024]. No contexto desta tese, essa ferramenta foi utilizada para agilizar a escrita e a depuração de trechos de código empregados nas simulações, bem como para automatizar ajustes estruturais em *scripts* complexos. Sua aplicação permitiu a redução do tempo gasto em tarefas repetitivas, possibilitando maior foco na análise dos resultados experimentais e na formulação das estratégias de otimização. É importante ressaltar que o *JetBrains AI assistant* foi empregado estritamente como um recurso auxiliar, sem influência na concepção dos conceitos e soluções propostas neste trabalho. Seu uso esteve restrito a melhorias operacionais, garantindo que todas as decisões técnicas e metodológicas fossem fundamentadas em princípios científicos e desenvolvidas pelo autor.

Escrita, Análise e refatoração de códigos

Correção de código

Durante o desenvolvimento e depuração dos *scripts* utilizados nas simulações desta tese, foi necessário identificar e corrigir inconsistências na manipulação de nomes de arquivos, particularmente na extração de parâmetros embutidos nas cadeias de texto. Para auxiliar nessa tarefa, recorreu-se ao *JetBrains AI Assistant*, que forneceu sugestões pontuais de análise e refatoração do código. A seguir, apresenta-se o comando utilizado para solicitar a identificação do problema e a sugestão de correção correspondente.

Comando: “Análise o trecho de código e encontre o problema na extração dos parâmetros dos nomes dos arquivos, destaque os problemas e uma proposta de solução”. A Listagem A.3 apresenta o código original apresentado para análise.

Listagem A.3: Código anexado ao comando no AI Assistant

```
# Loop pelas linhas do arquivo de entrada
while IFS= read -r line; do
    # Extraíndo os valores de <prefix>, <seed>, <gateways> e <
    devices> da linha
    echo $line
    if [[ $line =~ ^([^\s/]+)/\1_QoS_([0-9]+)s_([0-9]+)x1Gv
    _([0-9]+)D\.dat$ ]]; then
        line_prefix="${BASH_REMATCH[1]}"
        seed="${BASH_REMATCH[2]}"
        gateways="${BASH_REMATCH[3]}"
```

```

devices="${BASH_REMATCH[4]}"
echo "Executando simulação para $line_prefix, seed=$seed
    , gateways=$gateways, devices=$devices"
# Verifica se a linha atende aos critérios de seleção
if [[ "$line_prefix" == "$prefix" && "$gateways" -ge "$
minGateways" && "$gateways" -le "$maxGateways" && "$
devices" -ge "$minDevices" && "$devices" -le "$
maxDevices" ]]; then
#
    echo "Executando simulação para $line_prefix, seed
=$seed, gateways=$gateways, devices=$devices"
    # Executa a simulação com os valores extraídos
    echo "python3 04_biobj_model-gurobi.py "$gateways" 1
        "$devices" "$seed" $qos $alpha $beta 2>> errors.
        dat"
    fi
else
    echo "Linha ignorada (formato incompatível): $line"
fi
done < "$input_file"
echo "Execução das simulações concluída."

```

Resultado: A Listagem A.4 demonstra o resultado do *JetBrains AI Assistant*.

Listagem A.4: *Explicação do problema e sugestão de correção do
JetBrains AI Assistant*

```

Problema com a expressão regular:
- A regex usada no script pode estar incorreta ou não funcionar
  como esperado:
``` bash
 if [[$line =~ ^([^\s/]+)/\1_QoS_([0-9]+)s_([0-9]+)x1Gv
 _([0-9]+)D\.dat$]]; then
```
O problema está no uso de \1, que não é suportado pelo [[ ... =~
... ]] do Shell Script. No bash, ao usar regex dessa forma,
o backreference \1 deve ser substituído conforme necessário
para grupo de captura direto e $BASH_REMATCH deve ser
utilizado para validar os grupos.
- Correção sugerida: Caso o objetivo seja validar pelo prefixo,
  a regex pode ser corrigida para:
``` bash
 if [[$line =~ ^([^\s/+)/([^\s/]+)/QoS_([0-9]+)s_([0-9]+)x1Gv
 _([0-9]+)D\.dat$]]; then

```

```
'''
```

## Geração, análise e refatoração de código

Durante o processo de análise dos dados gerados pelas simulações, foi necessário desenvolver um *script* que automatizasse a leitura dos arquivos resultantes e a geração de gráficos comparativos com medidas estatísticas. Para esse fim, utilizou-se o *JetBrains AI Assistant* como apoio à escrita inicial do código, com base em parâmetros fornecidos manualmente e estruturados de acordo com o modelo de experimentação. O comando a seguir descreve a solicitação realizada à ferramenta para gerar o esqueleto funcional do *script*, incorporando estruturas de repetição, leitura de arquivos e visualização gráfica com intervalo de confiança.

### Comando:

#### Listagem A.5: Comando para geração de código no JetBrains AI Assistant

```
Crie um código para ler os arquivos na pasta `"/home/rogerio/git
/ns-allinone-3.42/ns-3.42/scratch/ql-uav-deployment/data/res
/<method>` com a estrutura de campos "seed,gwid,delay,
throughput,qos" e nomenclatura de arquivos: '<method>
QoSPerGw<seed>s_<virtual_position>V_<gateway>Gx<device>D.
dat', onde
```python
methods = ["a2c", "ppo", "dqn", "ddqn"]
seeds = [1, 2, 3, 4]
n_virtual_positions = [49, 64, 81, 100, 121, 144]
n_gateways = [2, 4]
n_devices = [50, 100, 150, 200]
'''

Gere dois gráficos com três subgráficos lado a lado cada
contendo os resultados das médias de atraso <delay>, vazão<
throughput> e QoS<qos>. O primeiro gráfico apresentará no
eixo x as posições candidatas <virtual_position> e o segundo,
o número de dispositivos <devices>. Cada subgráfico
apresentará <gateways> linhas para cada método <method>.
Apresente os resultados com intervalo de confiança de 95%
usando "matplotlib errorBar".
Ajuste os labels das linhas com:
```python
labels = {'a2c': 'NSE-A2C',
 'ppo': 'RG2E-PPO',
```

```

 'dqn': 'SR-DQN',
 'ddqn': 'DA-DDDQN'}
'''
seguido do número de gateways, no formato f"{labels[method]} {
 gateway} GW".
Ajuste os tamanhos dos rótulos com:
'''python
title_size = 20
label_size = 18
tick_size = 16
'''
Ajuste as cores das linhas com:
'''python
colors = {
 "DQN": "tab:red",
 "DDQN": "tab:blue",
 "PPO": "tab:green",
 "A2C": "tab:orange"
}
'''

```

A partir do comando fornecido e das instruções detalhadas sobre a estrutura dos dados e os objetivos da análise, o assistente de inteligência artificial gerou automaticamente um código completo em linguagem Python. Esse código tem como finalidade ler os arquivos de simulação, calcular as médias das métricas de desempenho (atraso, vazão e QoS) e apresentar os resultados por meio de gráficos com intervalos de confiança de 95%. A seguir, apresenta-se o script original gerado pela ferramenta.

### Resultado:

#### **Listagem A.6:** *Resultado da ferramenta JetBrains AI Assistant para solicitação de geração de código*

```

import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import sem, t

def plot_qos_metrics(base_path, methods, seeds,
 virtual_positions, devices, gateways):
 """

```

```
Lê os arquivos e gera gráficos para as métricas de delay,
throughput e QoS.

:param base_path: Caminho base da pasta contendo os arquivos
.
:param methods: Lista de métodos (por exemplo, ["a2c", "ppo",
"dqn", "ddqn"]).
:param seeds: Lista de seeds usadas nos arquivos.
:param virtual_positions: Lista de números de posições
virtuais.
:param devices: Lista de números de dispositivos.
:param gateways: Lista de números de gateways.
"""

Configurações
labels = {'a2c': 'NSE-A2C',
 'ppo': 'RG2E-PPO',
 'dqn': 'SR-DQN',
 'ddqn': 'DA-DDDQN'}

colors = {
 "dqn": "tab:red",
 "ddqn": "tab:blue",
 "ppo": "tab:green",
 "a2c": "tab:orange"
}

title_size = 20
label_size = 18
tick_size = 16

Função para calcular intervalo de confiança de 95%
def confidence_interval(data):
 if len(data) > 1:
 return sem(data) * t.ppf((1 + 0.95) / 2, df=len(data)
 - 1)
 else:
 return 0

Preparar armazenamento de resultados
results_vp = {metric: {method: {g: [] for g in gateways} for
 method in methods} for metric in
 ["delay", "throughput", "qos"]}
results_devices = {metric: {method: {g: [] for g in gateways
```

```
 } for method in methods} for metric in
 ["delay", "throughput", "qos"]}

Iterar pelos métodos e processar os arquivos
for method in methods:
 for g in gateways:
 # Analisar por virtual_positions
 for v in virtual_positions:
 metric_data = {metric: [] for metric in ["delay
 ", "throughput", "qos"]}
 for s in seeds:
 for d in devices:
 file_name = f"{method}_QoSPerGw_{s}s_{v}.
 V_{g}Gx{d}D.dat"
 file_path = os.path.join(base_path,
 method, file_name)
 if os.path.exists(file_path):
 # Ler o arquivo
 df = pd.read_csv(file_path, names=["
 seed", "gwid", "delay", "
 throughput", "qos"])
 metric_data["delay"].append(df["
 delay"].mean())
 metric_data["throughput"].append(df
 ["throughput"].mean())
 metric_data["qos"].append(df["qos"].
 mean())

 # Fazer média para cada metric
 for metric in metric_data:
 results_vp[metric][method][g].append(np.mean
 (metric_data[metric]) if metric_data[
 metric] else 0)

Analisar por dispositivos
for d in devices:
 metric_data = {metric: [] for metric in ["delay
 ", "throughput", "qos"]}
 for s in seeds:
 for v in virtual_positions:
 file_name = f"{method}_QoSPerGw_{s}s_{v}.
 V_{g}Gx{d}D.dat"
```

```
 file_path = os.path.join(base_path,
 method, file_name)
 if os.path.exists(file_path):
 # Ler o arquivo
 df = pd.read_csv(file_path, names=["
 seed", "gwid", "delay", "
 throughput", "qos"])
 metric_data["delay"].append(df["
 delay"].mean())
 metric_data["throughput"].append(df
 ["throughput"].mean())
 metric_data["qos"].append(df["qos"].
 mean())

 # Fazer média para cada metric
 for metric in metric_data:
 results_devices[metric][method][g].append(
 np.mean(metric_data[metric]) if
 metric_data[metric] else 0)

Criar gráficos
fig, axes = plt.subplots(2, 3, figsize=(20, 14))
metrics = ["delay", "throughput", "qos"]
titles = ["Atraso (ms)", "Vazão (Mbps)", "QoS"]

Gráficos por Virtual Positions
for i, metric in enumerate(metrics):
 ax = axes[0, i]
 for method in methods:
 for g in gateways:
 values = results_vp[metric][method][g]
 ci = [confidence_interval(values)] * len(
 virtual_positions) if values else [0] * len(
 virtual_positions)
 ax.errorbar(virtual_positions, values, yerr=ci,
 label=f"{labels[method]} {g} GW",
 color=colors[method], fmt="o-")
 ax.set_title(f"Média de {titles[i]} x Posições Virtuais
 ", fontsize=title_size)
 ax.set_xlabel("Posições Virtuais (v)", fontsize=
 label_size)
 ax.set_ylabel(titles[i], fontsize=label_size)
```

```
ax.tick_params(axis='both', which='major', labelsize=
 tick_size)
ax.legend(fontsize=tick_size)
ax.grid()

Gráficos por Dispositivos
for i, metric in enumerate(metrics):
 ax = axes[1, i]
 for method in methods:
 for g in gateways:
 values = results_devices[metric][method][g]
 ci = [confidence_interval(values)] * len(devices
) if values else [0] * len(devices)
 ax.errorbar(devices, values, yerr=ci, label=f"{
 labels[method]} {g} GW",
 color=colors[method], fmt="o-")
 ax.set_title(f"Média de {titles[i]} x Dispositivos",
 fontsize=title_size)
 ax.set_xlabel("Dispositivos (d)", fontsize=label_size)
 ax.set_ylabel(titles[i], fontsize=label_size)
 ax.tick_params(axis='both', which='major', labelsize=
 tick_size)
 ax.legend(fontsize=tick_size)
 ax.grid()

plt.tight_layout()
plt.show()

Parâmetros
methods = ["a2c", "ppo", "dqn", "ddqn"]
seeds = [1, 2, 3, 4]
n_virtual_positions = [49, 64, 81, 100, 121, 144]
n_gateways = [2, 4]
n_devices = [50, 100, 150, 200]
caminho_dados = "/home/rogerio/git/ns-allinone-3.42/ns-3.42/
 scratch/ql-uav-deployment/data/res"

Chamar a função para gerar os gráficos
plot_qos_metrics(caminho_dados, methods, seeds,
 n_virtual_positions, n_devices, n_gateways)
```

A execução inicial do código gerado resultou em uma falha durante a etapa de agregação estatística dos dados extraídos dos arquivos de simulação. O erro ocorreu especificamente ao tentar calcular a média dos valores de atraso (delay), evidenciando um problema na interpretação e formatação dos dados lidos. O trecho abaixo apresenta a mensagem de erro retornada pelo ambiente de execução, que motivou a análise posterior realizada com apoio do assistente de inteligência artificial.

### Erro apresentado na compilação:

#### **Listagem A.7:** *Solicitação de correção de erro de código no Jet-Brains AI Assistant a partir da interface da IDE*

```
Please help me understand what the problem is and try to fix
the code.
Console output:
'''
Traceback (most recent call last):
 File "/home/rogerio/git/A2C-LoVQI/ppo/QoS_Metrics_Graphs.py",
 line 135, in <module>
 plot_qos_metrics(caminho_dados, methods, seeds,
 n_virtual_positions, n_devices, n_gateways)
 File "/home/rogerio/git/A2C-LoVQI/ppo/QoS_Metrics_Graphs.py",
 line 60, in plot_qos_metrics
 metric_data["delay"].append(df["delay"].mean())
 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
 File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/series.py", line 6549, in mean
 return NDFrame.mean(self, axis, skipna, numeric_only, **
 kwargs)
 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
 File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/generic.py", line 12420, in mean
 return self._stat_function(
 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
 File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/generic.py", line 12377, in
 _stat_function
 return self._reduce(
 ^^^^^^^^^^^^^^^^^
 File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/series.py", line 6457, in _reduce
```

```

return op(delegate, skipna=skipna, **kwds)
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/nanops.py", line 147, in f
result = alt(values, axis=axis, skipna=skipna, **kwds)
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/nanops.py", line 404, in new_func
result = func(values, axis=axis, skipna=skipna, mask=mask,
**kwargs)
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/nanops.py", line 720, in nanmean
the_sum = _ensure_numeric(the_sum)
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "/home/rogerio/git/A2C-LoVQI/tf/lib/python3.12/site-
packages/pandas/core/nanops.py", line 1701, in
_ensure_numeric
raise TypeError(f"Could not convert string '{x}' to numeric
")
TypeError: Could not convert string 'delay0.2911320.277204' to
numeric

Process finished with exit code 1
'''

```

Após a execução do código gerado com apoio do assistente, foi identificado um erro na etapa de cálculo das médias dos parâmetros extraídos dos arquivos simulados. Para diagnosticar e propor uma solução apropriada, solicitou-se ao assistente de inteligência artificial uma análise detalhada do trecho de código responsável pela leitura e processamento das colunas numéricas. O retorno da ferramenta está descrito a seguir.

### **Análise do assistente de AI:**

#### **Listagem A.8:** *Explicação do erro de código do JetBrains AI Assistant*

O erro no console foi causado por uma tentativa de calcular a média de uma coluna no DataFrame `"delay"`, mas alguns valores nesta coluna foram interpretados como strings que não podem ser convertidos para números. A mensagem de erro indica que há um problema com um valor como `'delay0.2911320.277204'`, que parece ser uma concatenação

inadequada de múltiplos valores.

Vamos abordar o problema passo a passo para identificar e corrigi-lo.

### Causa do Problema

1. O erro acontece aqui no código:

```
``` python
metric_data["delay"].append(df["delay"].mean())
```
```

1. O campo 'delay' no DataFrame provavelmente contém strings em vez de números. Especificamente, houve algum problema no arquivo de dados ('.dat'), onde a separação entre colunas parece ter sido quebrada ou mal processada.
2. Como o arquivo do qual o DataFrame 'df' é carregado ('pd.read\_csv') não foi processado corretamente, a coluna 'delay' contém valores inválidos que misturam números com outros textos ('delay0.2911320.277204').

### Solução

A solução envolve identificar e corrigir o problema ao carregar os arquivos. Isso pode ser feito garantindo que:

1. O arquivo de entrada seja lido corretamente, usando o delimitador correto no 'pd.read\_csv'.
2. Convertamos as colunas esperadas para os tipos adequados (float, int, etc.) e descartemos quaisquer valores inválidos.

Aqui está uma solução detalhada:

### Correção Proposta

1. Garantir que o arquivo seja carregado corretamente usando 'pd.read\_csv'. Especificar o delimitador apropriado (por exemplo, 'delimiter="\t"', ',', ou outro).
2. Limpar os dados da coluna 'delay' (e outras, se necessário) para garantir que contenham apenas valores numéricos.
3. Ignorar ou lançar um aviso para valores inválidos que não podem ser convertidos para float.

Vou atualizar o código na parte de leitura e tratamento de arquivos para corrigir esses problemas: