

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA

GUSTAVO COELHO DOMINGOS

Controle adaptativo de pesos de QoS e equidade no escalonamento de downlink 5G via
aprendizado por reforço profundo

Uberlândia
2026

GUSTAVO COELHO DOMINGOS

Controle adaptativo de pesos de QoS e equidade no escalonamento de downlink 5G via
aprendizado por reforço profundo

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação em Engenharia
Elétrica da Universidade Federal de Uberlândia
como requisito parcial para obtenção do título
de mestre em Engenharia Elétrica.

Área de concentração: Processamento da
Informação.

Orientador: Dr. Ederson Rosa da Silva.

Uberlândia

2026

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU
com dados informados pelo(a) próprio(a) autor(a).

D671 Domingos, Gustavo Coelho, 1992-
2026 Controle adaptativo de pesos de QoS e equidade no
escalonamento de downlink 5G via aprendizado por reforço
profundo [recurso eletrônico] / Gustavo Coelho Domingos. - 2026.

Orientador: Ederson Rosa da Silva.
Dissertação (Mestrado) - Universidade Federal de Uberlândia,
Pós-graduação em Engenharia Elétrica.
Modo de acesso: Internet.
DOI <http://doi.org/10.14393/ufu.di.2026.652>
Inclui bibliografia.

1. Engenharia elétrica. I. Silva, Ederson Rosa da, 1984-, (Orient.).
II. Universidade Federal de Uberlândia. Pós-graduação em
Engenharia Elétrica. III. Título.

CDU: 621.3

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:
Gizele Cristine Nunes do Couto - CRB6/2091
Nelson Marcos Ferreira - CRB6/3074



ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Engenharia Elétrica				
Defesa de:	Dissertação de Mestrado, 830, PPGEELT				
Data:	Vinte e cinco de setembro de dois mil e vinte e seis	Hora de início:	8:30	Hora de encerramento:	10:35
Matrícula do Discente:	12422EEL004				
Nome do Discente:	Gustavo Coelho Domingos				
Título do Trabalho:	Controle adaptativo de pesos de QoS e equidade no escalonamento de downlink 5G via aprendizado por reforço profundo				
Área de concentração:	Processamento da Informação				
Linha de pesquisa:	Processamento Digital de Sinais e Redes de Comunicação				
Projeto de Pesquisa de vinculação:	Coordenador do projeto: Éderson Rosa da Silva. Título do Projeto: Desenvolvimento e simulação de técnicas de alocação de recursos em redes de comunicação. Agência financiadora: Não se aplica. Número do processo na agência financiadora: Não se aplica. Vigência do projeto: 2018-atual.				

Reuniu-se através de videoconferência, a Banca Examinadora designada pelo Colegiado do Programa de Pós-graduação em Engenharia Elétrica, assim composta:

Doutores: Keiji Yamanaka (UFU), Itagildo Edmar Garbazza (IFMG) e Éderson Rosa da Silva, coorientador do discente.

Iniciando os trabalhos o presidente da mesa, Prof. Dr. Éderson Rosa da Silva, apresentou a Comissão Examinadora e o candidato, agradeceu a presença do público, e concedeu ao discente a palavra para a exposição do seu trabalho. A duração da apresentação do discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor presidente concedeu a palavra, pela ordem sucessivamente, aos examinadores, que passaram a arguir o candidato. Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o candidato:

APROVADO.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Mestre. O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme, foi assinada pela Banca Examinadora.



Documento assinado eletronicamente por **Ederson Rosa da Silva, Professor(a) do Magistério Superior**, em 25/09/2026, às 10:37, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **ITAGILDO EDMAR GARBAZZA, Usuário Externo**, em 25/09/2026, às 10:37, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Keiji Yamanaka, Professor(a) do Magistério Superior**, em 25/09/2026, às 10:37, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **7647361** e o código CRC **AA1FB2D0**.

GUSTAVO COELHO DOMINGOS

Controle adaptativo de pesos de QoS e equidade no escalonamento de downlink 5G via
aprendizado por reforço profundo

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação em Engenharia
Elétrica da Universidade Federal de Uberlândia
como requisito parcial para obtenção do título
de mestre em Engenharia Elétrica.

Área de concentração: Processamento da
Informação.

Uberlândia, 25 de Setembro de 2026

Banca Examinadora:

Dr. Éderson Rosa da Silva – UFU

Dr. Keiji Yamanaka – UFU

Dr. Itagildo Edmar Garbaza – IFMG

Dedico este trabalho a Deus, que me sustentou
nos momentos em que eu não teria sustentado
a mim mesmo.

AGRADECIMENTOS

Aos meus pais, Antônio e Mariângela, por terem sustentado com trabalho e afeto todas as condições que me trouxeram até aqui, e por nunca terem tratado o estudo como algo dispensável. Às minhas irmãs, Daniela e Dra. Andreia, pelo companheirismo de sempre.

Às minhas avós, Idalira Maria e Sônia Costa, que partiram no decorrer deste mestrado. Devo a elas parte do que sou, e me faltou tempo para lhes mostrar aonde isso me levou. Gostaria que estivessem aqui para ver esta conquista, que também é delas. Sinto falta de poder colocar este trabalho em suas mãos.

Ao Me. Fábio Romero, pelo apoio constante e pelo companheirismo em todos os momentos desta caminhada. Soube como poucos entender as dificuldades de cada etapa, e a sua presença foi decisiva nos períodos mais difíceis.

Agradeço ao Prof. Dr. Éderson Rosa da Silva, amigo e orientador, pelo incentivo, pela motivação, pela paciência e pela dedicação nesta caminhada acadêmica.

Aos colegas Getúlio e Gabriel, pela disponibilidade e ajuda.

A Andressa, pela paciência, orientações e apoio durante essa minha jornada.

Agradeço a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa concedida durante os anos do curso.

“It's a dangerous business, Frodo, going out of
your door," he used
to say." You step into the Road, and if you
don't keep your feet,
there is no knowing where you might be swept
off to. ”

(TOLKIEN, 1954, p. 82)

RESUMO

O escalonamento de recursos no enlace de *downlink* de redes 5G com tráfego misto exige equilibrar três objetivos conflitantes: vazão agregada, equidade na distribuição e cumprimento de prazo do tráfego sensível à latência. Os escalonadores convencionais otimizam um eixo em detrimento dos demais, e as propostas orientadas a qualidade de serviço adotam pesos fixados em tempo de projeto, que não acompanham a variação das condições da rede. Este trabalho apresenta o DRLQoS Scheduler, um escalonador de enlace de descida cujos pesos de urgência de latência e de confiabilidade e cujo alvo de equidade são ajustados em tempo de execução por um agente treinado com o algoritmo *Deep Deterministic Policy Gradient*. Em vez de alocar blocos de recursos diretamente, o agente aplica controle paramétrico indireto sobre uma métrica proporcional-justa ponderada por qualidade de serviço, o que preserva a interpretabilidade da alocação e dispensa a otimização por slot exigida pelas propostas convencionais. A alocação é obtida em forma fechada, com complexidade linear no produto entre o número de blocos de recursos e o número de usuários. O escalonador foi implementado no Vienna 5G System Level Simulator e avaliado em uma rede heterogênea composta por três tipos de célula, em quatro cenários que variam densidade de usuários, mobilidade e prazo de tráfego sensível à latência, com 30 execuções independentes por configuração. Nas 36 comparações por teste t bilateral contra três linhas de base nativas, a proposta nunca é superada com significância estatística e prevalece em 25, com o cumprimento de prazo favorável nas 12 comparações dessa métrica. Sob saturação, com oito vezes a densidade de referência e topologia inalterada, é o único dos cinco escalonadores avaliados a ocupar a segunda posição nas três métricas simultaneamente.

Palavras-chave: aprendizado por reforço profundo; equidade; escalonamento de recursos; qualidade de serviço; redes 5G; redes heterogêneas.

ABSTRACT

Downlink resource scheduling in 5G networks carrying mixed traffic requires balancing three conflicting objectives: aggregate throughput, fairness, and deadline compliance for latency-sensitive flows. Conventional schedulers optimize one axis at the expense of the others, while quality-of-service (QoS)-aware designs rely on weights fixed at design time that do not track varying network conditions. This work presents the DRLQoSScheduler, a downlink scheduler whose latency and reliability urgency weights and fairness target are tuned at run time by an agent trained with the deep deterministic policy gradient (DDPG) algorithm. Rather than allocating resource blocks directly, the agent applies indirect parametric control to a QoS-weighted proportional fair metric, preserving the interpretability of the allocation and avoiding the per-slot optimization required by conventional approaches. The allocation is computed in closed form, with complexity linear in the product of the number of resource blocks and the number of users. The scheduler was implemented in the Vienna 5G System Level Simulator and evaluated on a three-tier heterogeneous network across four scenarios varying user density, mobility, and the latency-sensitive traffic deadline, with 30 independent runs per configuration. Across 36 two-tailed t-tests against three native baselines, the proposed scheduler is never outperformed with statistical significance and prevails in 25, with deadline compliance favorable in all 12 comparisons of that metric. Under saturation, at eight times the reference density with unchanged topology, it is the only one of the five evaluated schedulers ranking second in all three metrics simultaneously.

Index Terms—5G networks, deep reinforcement learning, fairness, heterogeneous networks, quality of service, resource scheduling.

LISTA DE ILUSTRAÇÕES

Figura 3.1 - Arquitetura da NG-RAN e 5GC	29
Figura 3.2 - Estrutura temporal NR	31
Figura 3.3 - Coexistência entre eMBB e uRLLC.	35
Figura 3.4- Framework de QoS do 5G (QoS Flow/ 5QI/ PDB/ PER).	37
Figura 3.5 - Arquitetura HetNet (Rede Heterogênea de 3 camadas).....	40
Figura 3.6 - Evolução dos escalonadores de Downlink.....	45
Figura 4.1- Loop de aprendizado por reforço.....	51
Figura 4.2 - Arquitetura da Deep Q-Network (DQN)	54
Figura 4.3- Arquitetura Ator-Crítico Off-Policy DDPG	58
Figura 4.4- Estratégia de Ensemble de Agentes DDPG	63
Figura 5.1 - Arquitetura do sistema proposto: laço fechado agente–ambiente–escalonador ...	68
Figura 5.2 - Disposição espacial das estações-base na topologia heterogênea multicamada...	87
Figura 5.3 - Curvas de recompensa por episódio dos dez agentes treinados.....	94
Figura 6.1 - Recompensa por episódio durante o treinamento dos dez agentes.....	103
Figura 6.2 - Ranqueamento dos agentes por recompensa média em avaliação sem ruído de exploração, com destaque para os cinco que compõem o ensemble	105
Figura 6.3 - Desempenho dos cinco escalonadores no cenário de saturação	114
Figura 6.4 - Matriz de significância das 36 comparações	118
Figura 6.5 - Planos de compromisso: vazão versus equidade e vazão versus cumprimento de prazo	119
Figura 6.6 - Comparação por métrica e cenário, com intervalos de confiança de 95%	120
Figura 6.7 - Dispersão das métricas entre as 30 rodadas independentes.....	121

LISTA DE TABELAS

Tabela 2.1 - Síntese comparativa dos principais trabalhos do estado da arte.....	23
Tabela 3.1- Requisitos mínimos de desempenho das classes de serviço 5G (ITU-R, 2017) ...	27
Tabela 3.2 - Parâmetros de numerologia do 5G NR (3GPP TS 38.211, 2020).....	30
Tabela 3.3 - Subconjunto de valores 5QI padronizados pelo 3GPP TS 23.501	37
Tabela 3.4 - Características das camadas na arquitetura HetNet.....	40
Tabela 3.5 - Classificação dos algoritmos de escalonamento em cenários 5G	44
Tabela 4.1 - Comparação entre abordagens de DRL para escalonamento em redes sem fio...	65
Tabela 5.1 - Responsabilidades dos módulos do sistema proposto.	69
Tabela 5.2 - Componentes do vetor de estado observado pelo agente.	73
Tabela 5.3 - Componentes do vetor de ação e mecanismos controlados.	74
Tabela 5.4 - Complexidade computacional por slot.	85
Tabela 5.5 - Parâmetros de radiofrequência e da grade de recursos.....	86
Tabela 5.6 - Camadas da rede heterogênea	87
Tabela 5.7 - Perfis de tráfego.....	88
Tabela 5.8 - Diferenças entre os cenários de treinamento e de avaliação.	89
Tabela 5.9 - Arquitetura das redes do ator e do crítico.....	91
Tabela 5.10 - Hiperparâmetros do agente e do treinamento.....	91
Tabela 5.11 - Protocolo de avaliação experimental.....	96
Tabela 6.1 - Cenários de avaliação	101
Tabela 6.2 - Resultado do treinamento dos dez agentes.....	104
Tabela 6.3 - Ranqueamento das políticas em avaliação sem exploração	105
Tabela 6.4 - Desempenho no cenário de referência	106
Tabela 6.5 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (desempenho no cenário de referência)	107
Tabela 6.6 - Cenário de carga elevada.....	110
Tabela 6.7 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (cenário de carga elevada).....	110
Tabela 6.8 - Cenário com mobilidade	111
Tabela 6.9 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (cenário com mobilidade).....	111
Tabela 6.10 - Cenário de estresse	112

Tabela 6.11 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (cenário de estresse)	112
Tabela 6.12 - Consolidação das 36 comparações estatísticas.....	117
Tabela 6.13 - Desempenho do Proportional Fair nos quatro cenários	122
Tabela 6.14 - DRLQoSScheduler comparado ao Proportional Fair	123
Tabela 6.15 - Tempo de inferência do ensemble de cinco políticas.....	125
Tabela 6.16 - Tempo de execução do escalonador por slot e por célula	126

LISTA DE ABREVIATURAS E SIGLAS

3GPP	3rd Generation Partnership Project
5QI	5G QoS Identifier
ABS	Almost Blank Subframes
AMC	Adaptive Modulation and Coding — adaptação de modulação e codificação
AMF	Access and Mobility Management Function
API	Application Programming Interface — interface de programação de aplicações
AR	Aprendizado por Reforço
Bi-LSTM	Bidirectional Long Short-Term Memory — memória longa de curto prazo bidirecional
BLER	Block Error Rate — taxa de erro de bloco
BSR	Buffer Status Report
BWP	Bandwidth Part
CNN	Convolutional Neural Network — rede neural convolucional
CoMP	Coordinated MultiPoint — transmissão multiponto coordenada
CP-OFDM	Cyclic Prefix Orthogonal Frequency Division Multiplexing
CQAS	Channel and QoS Aware Scheduler
CQI	Channel Quality Indicator
CRE	Cell Range Expansion
CSG	Closed Subscriber Group
CSI	Channel State Information
CUPS	Control and User Plane Separation
D2D	Device-to-Device
D3QN	Dueling Double Deep Q-Network
DCI	Downlink Control Information
DDDPG	Distributed Deep Deterministic Policy Gradient
DDPG	Deep Deterministic Policy Gradient
DFT-s-OFDM	Discrete Fourier Transform spread OFDM
DPG	Deterministic Policy Gradient
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning — aprendizado por reforço profundo

eICIC	enhanced Inter-Cell Interference Coordination
eMBB	Enhanced Mobile Broadband — banda larga móvel aprimorada
EXP/PF	Exponential/Proportional Fair
FFR	Fractional Frequency Reuse — reuso fracionário de frequência
FR	Frequency Range — faixa de frequência
FR1	Frequency Range 1
FR2	Frequency Range 2
GBR	Guaranteed Bit Rate
GFB	Guaranteed Flow Bit Rate
gNB	Next Generation NodeB
H-NOMA	Heterogeneous Non-Orthogonal Multiple Access
HARQ	Hybrid Automatic Repeat Request
HetNet	Heterogeneous Network — rede heterogênea
HOL	Head-of-Line
HTTP	Hypertext Transfer Protocol
ICIC	Inter-Cell Interference Coordination — coordenação de interferência entre células
IMT	International Mobile Telecommunications
IMT-2020	International Mobile Telecommunications 2020
IoT	Internet of Things — internet das coisas
ITU-R	International Telecommunication Union, Radiocommunication Sector
JT-CoMP	Joint Transmission Coordinated MultiPoint
KPI	Key Performance Indicator
LDPC	Low-Density Parity-Check
LPN	Low Power Node — nó de baixa potência
LTE	Long Term Evolution
LTE-A	LTE-Advanced
M-LWDF	Modified Largest Weighted Delay First
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
MDP	Markov Decision Process — processo de decisão de Markov
MFBR	Maximum Flow Bit Rate
MIMO	Multiple Input Multiple Output

mMTC	massive Machine Type Communications
NF	Network Function — função de rede
NFV	Network Function Virtualization
NG-RAN	Next Generation Radio Access Network
NOMA	Non-Orthogonal Multiple Access
Non-GBR	Non-Guaranteed Bit Rate
NP	Non-deterministic Polynomial time
NR	New Radio
NRT	Non-Real Time — tráfego não sensível a tempo
NSSF	Network Slice Selection Function
OFDM	Orthogonal Frequency Division Multiplexing
OSG	Open Subscriber Group
OU	Ornstein-Uhlenbeck
PA	Power Allocation — alocação de potência
PCF	Policy Control Function
PDB	Packet Delay Budget — orçamento de atraso de pacote
PDCP	Packet Data Convergence Protocol
PDU	Protocol Data Unit
PER	Packet Error Rate — taxa de erro de pacote
PF	Proportional Fair
PMI	Precoding Matrix Indicator — indicador de matriz de pré-codificação
POMDP	Partially Observable Markov Decision Process
PPO	Proximal Policy Optimization
PRB	Physical Resource Block
QAM	Quadrature Amplitude Modulation
QAS	QoS-Aware Scheduler
QFI	QoS Flow Identifier
QoS	Quality of Service — qualidade de serviço
QPSK	Quadrature Phase Shift Keying
RA	Resource Allocation — alocação de recursos
RAN	Radio Access Network
RB	Resource Block — bloco de recursos
RE	Resource Element

ReLU	Rectified Linear Unit — unidade linear retificada
RI	Rank Indicator — indicador de posto
RLC	Radio Link Control
RPPO	Recurrent Proximal Policy Optimization
RR	Round Robin
RRM	Radio Resource Management — gerenciamento de recursos de rádio
RSRP	Reference Signal Received Power
RT	Real Time — tráfego sensível a tempo
SARSA	State-Action-Reward-State-Action
SBA	Service-Based Architecture — arquitetura baseada em serviços
SDN	Software-Defined Networking
SIC	Successive Interference Cancellation — cancelamento sucessivo de interferência
SINR	Signal-to-Interference-plus-Noise Ratio
SLS	System Level Simulator
SMF	Session Management Function
SSB	Synchronization Signal Block — bloco de sinal de sincronização
TD	Temporal Difference — diferença temporal
TR	Technical Report — relatório técnico do 3GPP
TS	Technical Specification — especificação técnica do 3GPP
TTI	Transmission Time Interval
UA	User Association — associação de usuários
UE	User Equipment — equipamento de usuário
UPF	User Plane Function
uRLLC	Ultra-Reliable Low-Latency Communications
VoIP	Voice over Internet Protocol — voz sobre IP
WMMSE	Weighted Minimum Mean Square Error

SUMÁRIO

1	INTRODUÇÃO.....	12
1.1	Contextualização.....	12
1.2	Definição do Problema	13
1.3	Objetivos.....	13
1.3.1	Objetivo Geral	13
1.3.2	Objetivos Específicos	14
1.4	Justificativa	14
1.5	Contribuições	15
1.6	Metodologia	15
1.7	Organização do Trabalho	16
2	ESTADO DA ARTE	17
2.1	Método e Panorama da Revisão	17
2.2	Escalonamento Heurístico e Orientado a QoS.....	18
2.3	Coexistência eMBB/uRLLC e Multiplexação de Serviços	19
2.4	Aprendizado por Reforço Profundo para Escalonamento e RRM.....	20
2.4.1	Métodos Baseados em Valor	20
2.4.2	Métodos de Gradiente de Política e Controle Contínuo	21
2.4.3	Abordagens Multiagente e Observação Parcial	21
2.5	Equidade e Redes Heterogêneas.....	22
2.6	Síntese Comparativa e Lacuna de Pesquisa	22
2.7	Considerações Finais	25
3	REDES 5G E ESCALONAMENTO DE RECURSOS.....	27
3.1	Arquitetura e Casos de Uso do 5G	27
3.2	Estrutura de Transmissão (NR, TTI, RB).....	29
3.2.1	Frame, Subframe e Slot	30
3.2.2	Grade de Recursos, Resource Block e Resource Element.....	31
3.2.3	Adaptação de Enlace e Indicador de Qualidade do Canal	32
3.3	Tráfego Misto: eMBB e uRLLC.....	32
3.3.1	Modelos de Tráfego	33
3.3.2	Estratégias de Multiplexação	33
3.3.3	O Problema Multiobjetivo de Escalonamento	34
3.4	Qualidade de Serviço (QoS)	36

3.4.1	Framework de QoS do 5G	36
3.4.2	Fatores de Escalonamento Orientados a QoS	38
3.4.3	Métricas de Avaliação de QoS	38
3.5	Redes Heterogêneas (HetNets).....	39
3.5.1	Arquitetura Multicamada.....	39
3.5.2	Modos de Acesso e Associação de Usuários	41
3.5.3	Interferência em HetNets	41
3.5.4	RRM em HetNets e Relevância para Este Trabalho.....	42
3.6	Escalonamento de Pacotes em <i>Downlink</i>	43
3.6.1	Processo de Escalonamento	43
3.6.2	Limitações em Cenários com Tráfego Misto.....	43
3.7	Algoritmos de Referência (RR, <i>Best CQI</i>, QoS-Aware, PF)	44
3.7.1	Round Robin (RR).....	45
3.7.2	Best CQI	46
3.7.3	Proportional Fair (PF).....	46
3.7.4	M-LWDF e EXP/PF	47
3.7.5	QoS-Aware Scheduler (QAS/CQAS).....	47
3.8	Considerações Finais	49
4	APRENDIZADO POR REFORÇO PROFUNDO.....	50
4.1	Fundamentos de Aprendizado por Reforço	50
4.1.1	Processo de Decisão de Markov	50
4.1.2	Funções de Valor e Equações de Bellman.....	52
4.1.3	Q-Learning e Aprendizado Off-Policy	53
4.2	<i>Deep Q-Network (DQN)</i>	53
4.2.1	Experience Replay	54
4.2.2	Rede-Alvo.....	55
4.2.3	Double DQN e Dueling DQN.....	55
4.3	Métodos de Gradiente de Política	56
4.3.1	Teorema do Gradiente de Política	56
4.3.2	Gradiente de Política Determinístico	57
4.4	<i>Deep Deterministic Policy Gradient (DDPG)</i>.....	58
4.4.1	Estrutura do Algoritmo	58
4.4.2	Treinamento do Crítico.....	59
4.4.3	Atualização do Ator	59

4.4.4	Soft Update das Redes-Alvo.....	60
4.4.5	Exploração por Ruído de Ornstein-Uhlenbeck.....	60
4.5	Funções de Recompensa Multiobjetivo	61
4.6	<i>Ensemble</i> de Agentes em DRL	62
4.7	Aplicações em Redes sem Fio.....	63
4.8	Posicionamento do Presente Trabalho.....	64
4.9	Considerações Finais	65
5	SISTEMA PROPOSTO: DRLQOSSCHEDULER	67
5.1	Arquitetura do Sistema	68
5.2	Princípio de Projeto: Controle Paramétrico Indireto	69
5.2.1	Fundamentos para a Não Alocação Direta de RBs.....	70
5.2.2	Parametrização dos Compromissos Multiobjetivo	71
5.2.3	Espaço de Ação Contínuo e Escolha do Algoritmo.....	72
5.3	Formulação como Processo de Decisão de Markov	72
5.3.1	Espaço de Estados.....	72
5.3.2	Espaço de Ações	74
5.3.3	Função de Recompensa	75
5.3.4	Dinâmica de Transição e Fator de Desconto	76
5.4	O Escalonador DRLQoSScheduler	78
5.4.1	Métrica de Prioridade Proporcional-Justa Ponderada por QoS	78
5.4.2	Fator de Urgência de Latência	79
5.4.3	Fator de Urgência de Confiabilidade	81
5.4.4	Correção Iterativa de Equidade de Jain	82
5.4.5	Atualização da Memória de Vazão	84
5.4.6	Complexidade Computacional.....	84
5.5	Cenário de Simulação.....	85
5.5.1	Parâmetros de Radiofrequência e Grade de Recursos	85
5.5.2	Topologia Heterogênea Multicamada.....	86
5.5.3	Perfis de Tráfego.....	88
5.5.4	Cenários de Treinamento e de Avaliação	89
5.6	Arquitetura das Redes Neurais e Hiperparâmetros.....	90
5.6.1	Rede do Ator.....	90
5.6.2	Rede do Crítico	90
5.6.3	Hiperparâmetros de Treinamento	91

5.7	Protocolo de Treinamento Multiagente	92
5.8	Seleção de Agentes e Composição do Ensemble.....	94
5.9	Protocolo de Avaliação Experimental.....	95
5.10	Custo Computacional de Inferência.....	97
5.11	Limitações e Ameaças à Validade	97
5.12	Considerações Finais	99
6	AVALIAÇÃO DE DESEMPENHO	100
6.1	Protocolo Experimental Executado.....	100
6.2	Convergência do Treinamento	102
6.2.1	Curvas de Recompensa.....	102
6.2.2	Ranqueamento e Composição do Ensemble.....	104
6.3	Desempenho no Cenário de Referência	106
6.3.1	Vazão	107
6.3.2	Equidade	108
6.3.3	Cumprimento de Prazo	108
6.4	Generalização para Cenários Não Vistos	109
6.4.1	Carga Elevada.....	109
6.4.2	Mobilidade.....	111
6.4.3	Estresse	112
6.4.4	Comparabilidade entre Cenários.....	115
6.5	Análise Consolidada	117
6.6	Linha de Base Proportional Fair	121
6.7	Custo Computacional	124
6.8	Discussão.....	128
6.9	Considerações Finais	130
7	CONCLUSÃO.....	131
7.1	Síntese dos Resultados	131
7.2	Contribuições	133
7.3	Trabalhos Futuros	134
7.4	Considerações Finais	135
	REFERÊNCIAS	137

1 INTRODUÇÃO

1.1 Contextualização

A quinta geração de redes móveis (5G), padronizada pelo 3GPP no âmbito do programa *International Mobile Telecommunications* (IMT-2020), foi concebida para atender, sobre uma mesma infraestrutura, a três classes de serviço com requisitos mutuamente conflitantes: a banda larga móvel aprimorada (eMBB, do inglês *Enhanced Mobile Broadband*), orientada a altas taxas de transmissão; as comunicações ultraconfiáveis de baixa latência (uRLLC, do inglês *Ultra-Reliable Low-Latency Communications*), sensíveis a prazo (*deadline*) e a confiabilidade; e as comunicações massivas entre máquinas (mMTC, do inglês *massive Machine Type Communications*) (ITU-R, 2017). É na função de escalonamento de recursos de rádio, executada pela estação-base a cada intervalo de transmissão, que esses requisitos são conciliados ou violados: a cada decisão, o escalonador determina quais usuários recebem quais RBs, arbitrando de forma implícita o compromisso entre eficiência, equidade e atendimento a prazos.

Esse arbítrio é agravado por dois fatores. O primeiro é a coexistência de tráfego heterogêneo na mesma grade de tempo-frequência, em que fluxos elásticos de banda larga competem com fluxos esporádicos e sensíveis a prazo. O segundo é a densificação por redes heterogêneas (HetNets, do inglês *Heterogeneous Networks*), nas quais células de tipo macro, pico e femto operam com potências separadas por ordens de grandeza, ampliando a variabilidade espacial das condições de canal. O problema resultante é intrinsecamente multiobjetivo: maximizar a vazão agregada, preservar a equidade na distribuição de recursos e respeitar os prazos e a confiabilidade dos serviços críticos são metas que, sob recursos limitados, não podem ser satisfeitas simultaneamente em seu grau máximo.

As heurísticas clássicas de escalonamento, como o *Round Robin*, o *Best CQI*, o *Proportional Fair* (PF) e as variantes orientadas a Qualidade de Serviço (QoS, do inglês *Quality of Service*), abordam esse compromisso por meio de parâmetros fixados em tempo de projeto (MAMANE et al., 2022). Tal calibração confere ótimo desempenho sob as condições nominais para as quais foi ajustada, mas degrada-se quando a carga, a mobilidade ou a composição do tráfego se afastam desse ponto. Para superar essa rigidez, o aprendizado por reforço profundo (DRL, do inglês *Deep Reinforcement Learning*) consolidou-se como paradigma para o gerenciamento de recursos de rádio (ALWARAFY et al., 2022). A maioria das propostas, contudo, atribui ao agente a alocação direta dos RBs, o que incorre em um espaço de ações de

crescimento combinatório, dependente da configuração da rede, e abre mão das garantias analíticas das heurísticas consagradas (SEGUIN et al., 2023).

1.2 Definição do Problema

O problema que motiva esta dissertação decorre da constatação anterior: nenhuma ponderação fixa entre vazão, equidade, latência e confiabilidade é ótima em toda a faixa de operação de uma rede real, e a adaptação a condições variáveis exige o ajuste dinâmico do ponto de operação. A formulação mais direta desse ajuste por aprendizado por reforço, na qual o agente aloca diretamente os RBs, esbarra em três dificuldades simultâneas: a dimensionalidade combinatória e dependente da configuração do espaço de ações, o custo de inferência incompatível com o intervalo de decisão da interface aérea e a perda das garantias de comportamento que as heurísticas oferecem por construção.

Diante desse quadro, a questão de pesquisa que orienta o trabalho pode ser enunciada da seguinte forma: o controle paramétrico adaptativo de uma heurística de escalonamento orientada a QoS, exercido por um agente de aprendizado por reforço profundo, é capaz de produzir um escalonador equilibrado, que concilie a equidade na distribuição de recursos com o atendimento a requisitos de vazão e de prazo sob condições dinâmicas de rede, sem violar as restrições de tempo de execução típicas da camada de acesso? A hipótese assumida é a de que preservar a heurística de alocação e delegar ao agente apenas a escolha contínua de um pequeno conjunto de parâmetros de alto nível é suficiente para obter adaptabilidade sem sacrificar as garantias de previsibilidade e de custo computacional da heurística original.

1.3 Objetivos

1.3.1 Objetivo Geral

Propor, implementar e avaliar o DRLQoSScheduler, um escalonador de *downlink* para redes 5G heterogêneas cujos pesos de QoS e cuja meta de equidade são ajustados em tempo de execução por um agente de aprendizado por reforço profundo, de modo a conciliar vazão, equidade e cumprimento de prazo em cenários com tráfego misto eMBB/uRLLC.

1.3.2 Objetivos Específicos

Para alcançar o objetivo geral, definem-se os seguintes objetivos específicos: (i) formular o ajuste dos parâmetros do escalonador como um Processo de Decisão de Markov e resolvê-lo por meio do algoritmo *Deep Deterministic Policy Gradient* (DDPG), com espaço de ação contínuo de três dimensões, a saber, a meta de equidade de Jain, o peso de urgência de latência e o peso de confiabilidade; (ii) implementar o escalonador proposto sobre o simulador *Vienna 5G System Level Simulator* (SLS), preservando a heurística de alocação determinística e restringindo a atuação do agente ao controle paramétrico indireto; (iii) conceber um protocolo de treinamento multiagente, com sementes documentadas e composição de um *ensemble* de políticas, capaz de mitigar a variância inerente ao treinamento por DRL; (iv) estabelecer um protocolo experimental com respaldo estatístico, comparando o escalonador proposto às referências *Round Robin*, *Best CQI* e *QoS-Aware*, bem como a uma linha de base *Proportional Fair*; (v) avaliar a capacidade de generalização da política aprendida para cenários de rede não vistos durante o treinamento, sem qualquer reajuste; e (vi) caracterizar empiricamente o custo computacional do método, separando o agente do escalonador e situando-os em seus respectivos orçamentos temporais.

1.4 Justificativa

A relevância deste trabalho tem duas dimensões, uma científica e outra prática. Do ponto de vista científico, a revisão da literatura, apresentada no Capítulo 2, evidencia lacunas convergentes: a escassez de propostas em que o agente atua como controlador de alto nível de uma heurística de domínio, em vez de alocar recursos diretamente; o tratamento da equidade como objetivo explicitamente controlável, e não como subproduto da função de recompensa; a avaliação conjunta de tráfego eMBB/uRLLC sobre topologias heterogêneas multicamada; e o rigor de reprodutibilidade e de generalização, frequentemente comprometido pela dependência de execuções únicas e de simuladores fragmentados. A proposta posiciona-se precisamente na interseção dessas lacunas.

Do ponto de vista prático, a preservação da heurística de alocação mantém a previsibilidade e a eficiência computacional exigidas por um escalonador que decide a cada intervalo de transmissão, atributos indispensáveis à aplicabilidade em hardware de estação-base. Ao delegar ao aprendizado apenas o ajuste do ponto de operação, a abordagem alinha-se à visão de redes autônomas e nativamente inteligentes associada à evolução para a sexta geração

e às arquiteturas de rede de acesso aberta, nas quais controladores inteligentes ajustam continuamente o comportamento da rede sem substituir os mecanismos determinísticos que garantem a sua operação.

1.5 Contribuições

As principais contribuições desta dissertação são quatro. A primeira, de natureza conceitual, é a formulação do escalonamento adaptativo como um problema de controle paramétrico indireto, em que o agente ajusta três parâmetros contínuos de uma heurística orientada a QoS, mantendo a dimensão do espaço de ação independente do número de usuários e blocos e preservando as garantias de tempo de execução. A segunda, de natureza metodológica, é o isolamento da contribuição do aprendizado por meio de uma linha de base *Proportional Fair* obtida do próprio escalonador proposto com o controle desativado, procedimento imune à objeção de que diferenças de implementação contaminariam a comparação. A terceira é um protocolo experimental com respaldo estatístico, baseado em treinamento multiagente, *ensemble* de políticas, múltiplas rodadas independentes e reporte sistemático de intervalos de confiança e testes de significância. A quarta é a caracterização empírica do custo computacional do método, que separa o agente do escalonador, situa cada um em seu orçamento temporal e responde à objeção mais frequente ao emprego de aprendizado por reforço em decisões de camada de acesso.

Parte dos resultados desta dissertação foi aceita para apresentação no XLIV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, sob o título *Controle indireto por ensemble DDPG para escalonamento multiobjetivo em HetNets 5G com tráfego misto eMBB/uRLLC* (DOMINGOS; SILVA, 2026).

1.6 Metodologia

A investigação é conduzida por simulação de nível de sistema, empregando o Vienna 5G SLS versão 1.4 (MÜLLER et al., 2018), em conformidade com as recomendações do 3GPP. O escalonador proposto estende uma heurística sensível a QoS de referência (SHIYAHIN et al., 2022; QUEIROZ; SILVA, 2024) e é controlado por um agente DDPG que ajusta, a cada intervalo de decisão, os três parâmetros de alto nível. O cenário de avaliação é uma rede heterogênea de três camadas, com macro, pico e femtocélulas e tráfego misto eMBB/uRLLC. O treinamento produz dez agentes independentes, dos quais os cinco de melhor desempenho

compõem um *ensemble* de pesos uniformes. A avaliação abrange quatro cenários de rede, três deles ausentes do treinamento, com trinta rodadas independentes por combinação, e as comparações empregam testes de significância bilaterais com intervalos de confiança de 95%. O detalhamento completo da proposta e do protocolo consta dos Capítulos 5 e 6.

1.7 Organização do Trabalho

Além desta introdução, a dissertação organiza-se em seis capítulos. O Capítulo 2 apresenta o estado da arte do escalonamento de recursos em redes 5G e delimita a lacuna de pesquisa que o trabalho pretende preencher. O Capítulo 3 revisa os fundamentos das redes 5G e do escalonamento de recursos, incluindo a estrutura de transmissão, o tráfego misto eMBB/uRLLC, o arcabouço de QoS, as redes heterogêneas e os algoritmos de referência. O Capítulo 4 expõe as bases teóricas do aprendizado por reforço profundo, do Processo de Decisão de Markov ao algoritmo DDPG e à estratégia de *ensemble* de agentes. O Capítulo 5 detalha o sistema proposto, o DRLQoSScheduler, desde o princípio de projeto do controle paramétrico indireto até a arquitetura das redes neurais, o cenário de simulação e as limitações assumidas. O Capítulo 6 apresenta a avaliação experimental, com a análise de convergência, o desempenho no cenário de referência, a generalização para cenários não vistos, a comparação com a linha de base *Proportional Fair* e a caracterização do custo computacional. Por fim, o Capítulo 7 sintetiza os resultados, delimita as contribuições e projeta as direções de continuidade.

2 ESTADO DA ARTE

Este capítulo mapeia a produção científica sobre o escalonamento de pacotes e a alocação de recursos de rádio em redes 5G, com ênfase no problema central desta dissertação: o atendimento simultâneo de tráfego heterogêneo de banda larga móvel aprimorada e de comunicações ultraconfiáveis de baixa latência em HetNets, sob restrições de QoS e equidade. Busca-se apresentar o que já foi estudado, as metodologias predominantes e as principais conclusões da literatura, para então identificar a lacuna de conhecimento que a solução proposta, o DRLQoS Scheduler, pretende preencher.

2.1 Método e Panorama da Revisão

A revisão priorizou trabalhos publicados entre 2018 e 2026, recuperados majoritariamente do acervo IEEE *Xplore* e complementados por artigos de acesso aberto e por *surveys* de referência na área. O recorte temático restringiu-se a três eixos diretamente relacionados ao problema: o escalonamento de *downlink* e a alocação de recursos orientados a QoS em 5G *New Radio* (NR); a coexistência e a multiplexação de serviços eMBB e uRLLC; e a aplicação de aprendizado por reforço profundo (DRL) ao gerenciamento de recursos de rádio (RRM, do inglês *Radio Resource Management*). Trabalhos voltados exclusivamente a temas correlatos, porém distintos, como fatiamento de núcleo, *beamforming*, múltiplas entradas e múltiplas saídas em escala massiva (MIMO, do inglês *Multiple Input Multiple Output*), eficiência energética isolada ou arquiteturas de redes definidas por software (SDN, do inglês *Software-Defined Networking*) e de virtualização de funções de rede (NFV, do inglês *Network Function Virtualization*), foram considerados apenas quando ofereciam contribuições metodológicas transferíveis ao escalonamento na interface aérea.

Sob o aspecto quantitativo, o corpo principal analisado reúne cerca de duas dezenas de trabalhos primários, além de quatro *surveys* abrangentes (MAMANE *et al.*, 2022; ALWARAFY *et al.*, 2022; NGUYEN *et al.*, 2021; SEGUIN *et al.*, 2023). A distribuição temporal revela uma nítida aceleração da produção a partir de 2020, com concentração dos trabalhos baseados em DRL entre 2021 e 2024 e a presença de contribuições muito recentes, de 2026, o que confirma tratar-se de uma linha de pesquisa ativa e em plena maturação. Entre os veículos de publicação figuram periódicos de alto impacto, como o *IEEE Journal on Selected Areas in Communications*, as *IEEE/ACM Transactions on Networking*, as *IEEE Transactions on Wireless Communications*, as *IEEE Transactions on Mobile Computing* e as *IEEE Transactions*

on *Vehicular Technology*, bem como conferências consolidadas, entre elas GLOBECOM, ICC, PIMRC e MILCOM, o que atesta a relevância e o rigor do tema na comunidade científica.

Sob o aspecto qualitativo, observa-se o predomínio quase absoluto da avaliação por simulação, seja de nível de sistema (*System Level Simulation*), seja de nível de enlace (*Link Level Simulation*), com poucos trabalhos recorrendo a plataformas experimentais. Chama a atenção, ainda, a fragmentação das ferramentas de simulação, pois cada grupo tende a empregar simuladores próprios ou fortemente customizados, o que dificulta a comparação direta dos resultados. Nesse cenário, o simulador Vienna 5G SLS, de código aberto e em conformidade com o 3GPP, destaca-se como um dos poucos ambientes padronizados reutilizados por diferentes grupos, favorecendo a reprodutibilidade. As subseções seguintes organizam a literatura em quatro blocos temáticos, a saber, o escalonamento heurístico orientado a QoS, a multiplexação eMBB/uRLLC, o DRL aplicado ao escalonamento e ao RRM, e a equidade em HetNets, e conduzem à síntese comparativa e à delimitação da lacuna de pesquisa.

2.2 Escalonamento Heurístico e Orientado a QoS

A base histórica do escalonamento de pacotes assenta-se em algoritmos heurísticos de baixo custo computacional, capazes de operar dentro do rígido orçamento temporal de cada intervalo de transmissão. Mamane et al. (2022) oferecem o *survey* mais abrangente desse universo e propõem uma taxonomia que classifica os escalonadores em famílias orientadas à equidade, como o *Round Robin* e o *Proportional Fair*, ao canal, como o *Best CQI*, ao atraso, como o M-LWDF e o EXP/PF, e à QoS. A principal conclusão dos autores converge com a de outras revisões (ALWARAFY et al., 2022): os escalonadores clássicos operam com parâmetros fixos, calibrados para condições específicas, e por isso degradam-se quando a carga, a mobilidade ou a composição de tráfego se afastam do ponto de projeto.

Vários estudos procuraram aprimorar essas heurísticas incorporando de forma explícita requisitos de QoS. Prasad e Sunny (2024) formulam um problema de escalonamento de média temporal de longo prazo que minimiza uma função ponderada de pacotes descartados e apresentam políticas quase ótimas com garantia de atraso e isolamento entre fluxos. Shahid et al. (2026), por sua vez, propõem uma métrica de prioridade ciente de *deadline* que combina a eficiência espectral instantânea com o orçamento de atraso de pacote residual, distribuindo os blocos de recurso em dois estágios entre fluxos com taxa de bits garantida (GBR, do inglês *Guaranteed Bit Rate*) e non-GBR de modo a proteger prazos sem provocar a inanição dos fluxos de melhor esforço. Em ambos os casos, contudo, a lógica de priorização permanece estática.

São especialmente relevantes para esta dissertação os trabalhos que empregam o simulador Vienna 5G SLS em cenários heterogêneos. Mamane et al. (2019) analisam comparativamente esquemas de escalonamento de recursos de rádio para HetNets 5G nesse ambiente, com foco na melhoria de *throughput* e de relação sinal-interferência-ruído (SINR, do inglês *Signal-to-Interference-plus-Noise Ratio*). Queiroz e Silva (2024) propõem o *Channel and QoS Aware Scheduler* (CQAS), que estende o *QoS-Aware Scheduler* (QAS) ao integrar as condições de canal (CQI) ao processo de priorização. Avaliado em HetNet com múltiplos modelos de tráfego, incluindo *full buffer*, HTTP, veicular, VoIP, *gaming* e vídeo, o CQAS reporta ganhos de *throughput* de até 16,3% sobre o QAS e de 163,6% sobre o *Round Robin* para usuários de vídeo, mantendo as restrições de atraso das aplicações de tempo real. O QAS e o CQAS constituem justamente as linhas de base heurísticas orientadas a QoS sobre as quais o presente trabalho se apoia, e a sua principal limitação, os pesos de urgência fixos, motiva diretamente a abordagem adaptativa aqui proposta.

2.3 Coexistência eMBB/uRLLC e Multiplexação de Serviços

A coexistência de serviços com requisitos conflitantes na mesma grade de tempo-frequência é um dos problemas mais estudados da literatura recente. Popovski et al. (2018) fornecem a fundamentação teórico-informacional do fatiamento da rede de acesso para eMBB, uRLLC e mMTC, e demonstram que o compartilhamento não-ortogonal de recursos pode superar a alocação ortogonal em regimes específicos de carga, ao custo de interferência mútua. Anand et al. (2020) formalizam o escalonamento conjunto de eMBB e uRLLC em *downlink*, no qual o tráfego uRLLC é sobreposto de forma preemptiva, por *puncturing*, às alocações eMBB na escala de *mini-slot*. Os autores derivam um escalonador ótimo por gradiente iterativo para o modelo de perda linear e mostram que, para modelos de perda convexos ou por limiar, as decisões de eMBB e de uRLLC deixam de ser desacopláveis, o que exige otimização conjunta.

Em complemento a essa vertente, autores comparam, por otimização, as abordagens de numerologia mista e de *mini-slots* como mecanismos de multiplexação de baixa latência, maximizando a taxa dos usuários de melhor esforço sob restrição de latência. Liu et al. (2024) abordam o mesmo problema sob a ótica da equidade e evidenciam que a priorização do uRLLC tende a onerar de modo desproporcional os usuários eMBB de melhor canal, propondo um modelo explícito do compromisso entre taxa e *fairness*. Já Huang et al. (2022), com o sistema DELUXE, aplicam aprendizado profundo à adaptação de enlace, isto é, à seleção do esquema

de modulação e codificação para usuários eMBB submetidos a *puncturing*, alcançando predição de BLER em tempo real, inferior a 125 microssegundos, e demonstrando a viabilidade prática de técnicas de aprendizado no laço de decisão do NR. O conjunto dessas contribuições estabelece o arcabouço do problema, mas trata predominantemente de soluções analíticas ou de escopo restrito, como a adaptação de MCS, sem oferecer um controlador adaptativo do escalonador capaz de responder de forma dinâmica ao estado da rede.

2.4 Aprendizado por Reforço Profundo para Escalonamento e RRM

A limitação estrutural das heurísticas de parâmetros fixos motivou a adoção do aprendizado por reforço profundo como paradigma para o RRM. Os *surveys* de Alwarafy et al. (2022), sobre DRL para RRM em HetNets futuras, e de Nguyen et al. (2021) e Seguin et al. (2023), estes últimos voltados especificamente ao escalonamento de *downlink* com granularidade de bloco de recurso, convergem em três diagnósticos. Primeiro, o DRL lida naturalmente com a não-estacionariedade e a alta dimensionalidade dos ambientes reais. Segundo a maioria dos trabalhos ainda aloca blocos de recurso de forma direta, o que incorre em espaços de ação de crescimento combinatório. Terceiro, a reprodutibilidade e a generalização entre cenários permanecem como desafios em aberto. As subseções a seguir organizam os trabalhos por classe algorítmica.

2.4.1 Métodos Baseados em Valor

Os métodos baseados em valor, com destaque para a *Deep Q-Network* (DQN) e suas variantes, foram os primeiros a serem aplicados ao escalonamento. AlQwider et al. (2022) projetam um escalonador de *downlink* 5G NR fundamentado em DQN, cujo objetivo é maximizar o *throughput* da célula mantendo uma alocação justa, e demonstram, em simulação de nível de sistema parametrizada segundo as especificações do 3GPP, um melhor equilíbrio entre *throughput* e *fairness* do que o dos escalonadores tradicionais. Wang et al. (2024) propõem uma arquitetura de dois níveis para sistemas 5G de internet das coisas (IoT, do inglês *Internet of Things*) com tolerância a atraso, na qual uma *Dueling Double DQN* (D3QN) resolve a alocação de blocos entre fatias, apoiada por um preditor de tráfego que combina CNN e Bi-LSTM, enquanto uma heurística cuida da alocação intra-fatia. Para tratar o acoplamento entre blocos e potência, os autores adotam uma estrutura em ramos que desacopla o espaço de ação. Yu et al. (2023), por sua vez, aplicam DRL ao roteamento e ao escalonamento determinístico

de fluxos de criticalidade mista em redes industriais, com função de recompensa ciente de criticalidade e mascaramento de ações. O traço comum a esses trabalhos é a necessidade de discretizar o espaço de ação, o que limita a granularidade das decisões e escala mal em cenários densos.

2.4.2 Métodos de Gradiente de Política e Controle Contínuo

Para superar a discretização, uma segunda vertente recorre a métodos de gradiente de política, capazes de operar em espaços de ação contínuos. Hu et al. (2023) propõem a arquitetura híbrida DDPG-D3QN para redes colaborativas de borda e nuvem, combinando o DDPG para as variáveis contínuas, como a taxa de *offloading* e o controle de potência, com a D3QN para as decisões discretas, como a seleção de servidor, e reportam maior estabilidade e convergência mais rápida do que a de métodos DRL concorrentes. Budhiraja et al. (2021) integram DQN e DDPG em um esquema distribuído, o DDDPG, para controlar o escalonamento *Proportional Fair* em comunicação dispositivo a dispositivo (D2D, do inglês *Device-to-Device*) subjacente, com o objetivo explícito de melhorar a *fairness* entre enlaces. Stigenberg et al. (2021), com o escalonador QADRA, treinam um agente DRL ciente de QoS de forma fim a fim, otimizando diretamente a taxa de satisfação de QoS enquanto maximizam o desempenho da rede, e reportam ganho de 30% em *throughput* mantendo a satisfação de usuários VoIP. Esses trabalhos evidenciam a maturidade do controle contínuo, mas, em regra, o agente atua diretamente sobre a alocação de recursos, e não como um controlador de alto nível de uma heurística de domínio.

2.4.3 Abordagens Multiagente e Observação Parcial

Uma terceira vertente concentra-se na escalabilidade e no realismo da informação disponível. Naderializadeh et al. (2021) desenvolvem um arcabouço multiagente descentralizado no qual cada transmissor decide, de forma autônoma, a potência e o usuário a servir. O projeto dos espaços de observação e de ação é escalável e robusto a discrepâncias entre treino e teste, alcançando bom compromisso entre a taxa média e o 5º percentil. Chang et al. (2022) aplicam aprendizado por reforço multiagente ao escalonamento de *downlink* 5G NR sob realimentação de CQI e superam variantes do *Proportional Fair* com baixo tempo computacional. Hao et al. (2023) reconhecem de forma explícita a natureza parcialmente observável do problema, formulando-o como um POMDP e propondo um algoritmo *Recurrent*

Proximal Policy Optimization (PPO) capaz de operar sem informação completa de canal, com reduções expressivas no atraso de cauda. Kumbi et al. (2026) exploram a correlação entre a mobilidade e a qualidade de canal para projetar um escalonador de *downlink* baseado em DRL, ciente de mobilidade e de QoS, em ambientes de rodovia, reduzindo o consumo de energia em 18,2%. Esses trabalhos ampliam o realismo do modelo, mas nenhum deles combina, ao mesmo tempo, tráfego misto eMBB/uRLLC, topologia HetNet multicamada e controle adaptativo de uma heurística orientada a QoS.

2.5 Equidade e Redes Heterogêneas

A equidade na distribuição de recursos e a densificação por redes heterogêneas constituem temas transversais. Pratap et al. (2021) formulam a alocação conjunta de recursos, a minimização de interferência e a maximização de *fairness* em nível de usuário e de célula em HetNets 5G de *small cells* como um problema NP-difícil, e propõem algoritmos centralizado, distribuído e distribuído aleatorizado que superam esquemas de referência tanto em baixa quanto em alta densidade. Esse trabalho evidencia a complexidade combinatória do RRM em HetNets e reforça a relevância do índice de Jain como métrica de projeto, embora não recorra a aprendizado. Na interseção entre *fairness* e aprendizado, o já mencionado esquema DDDPG de Budhiraja et al. (2021) incorpora a equidade como objetivo, mas em cenário D2D, e não em HetNets multicamada com eMBB/uRLLC. De modo geral, verifica-se que a *fairness* raramente é tratada como restrição explícita e controlável dentro do laço de decisão dos escalonadores baseados em DRL, aparecendo, com mais frequência, como subproduto da função de recompensa.

2.6 Síntese Comparativa e Lacuna de Pesquisa

A Tabela 2.1 sintetiza os trabalhos mais representativos revisados e os confronta segundo a técnica empregada, o cenário e o tráfego considerados, os objetivos de QoS e de equidade, a forma de avaliação e a limitação mais relevante para o posicionamento desta dissertação.

Tabela 2.1 - Síntese comparativa dos principais trabalhos do estado da arte.

Trabalho	Técnica	Cenário / Tráfego	Objetivos (QoS / Fairness)	Avaliação	Limitação relevante a este trabalho
Popovski et al. (2018)	Análise teórico-informacional (slicing ortogonal vs. não-ortogonal)	eMBB/uRLLC/mMTC, <i>uplink</i>	Isolamento e garantias por fatia	Analítica	Não trata escalonamento adaptativo em <i>downlink</i>
Anand et al. (2020)	Escalonador conjunto por gradiente iterativo com puncturing	eMBB/uRLLC, <i>downlink</i>	Utilidade eMBB sob deadline uRLLC	Simulador não informado	Parâmetros fixos, sem aprendizado
Marijanović et al. (2021)	Otimização (numerologia mista vs. mini-slots)	Baixa latência vs. banda	Taxa sob restrição de latência	<i>Vienna 5G SLS</i>	Solução offline, não adaptativa
Liu et al. (2024)	Multiplexação de RB centrada em fairness	eMBB/uRLLC	Compromisso taxa/fairness	Simulador não informado	Sem controle contínuo aprendido
Huang et al. (2022): DELUXE	Aprendizado profundo para adaptação de enlace (MCS)	eMBB sob puncturing uRLLC	Confiabilidade e eficiência espectral	Simulação de nível de enlace	Foco em MCS, não em alocação de RB
Mamane et al. (2022)	Survey e taxonomia de escalonadores	5G, tráfego misto	QoS, <i>fairness</i> , <i>latência</i>	Revisão	Aponta a lacuna dos parâmetros fixos
Queiroz e Silva (2024): CQAS	Heurístico de canal e QoS (estende o QAS)	HetNet, múltiplos tráfegos	QoS, <i>fairness</i> , <i>throughput</i>	<i>Vienna 5G SLS</i>	Pesos fixos, sem adaptação dinâmica
AlQwider et al. (2022)	DQN (baseado em valor, ação discreta)	5G NR <i>downlink</i>	<i>Throughput</i> e <i>fairness</i>	Simulador não informado	Discretização limita a granularidade
Wang et al. (2024)	D3QN em dois níveis com previsão de tráfego	5G IIoT, <i>slicing</i>	QoS por fatia (atraso)	Simulador não informado	Ação discreta, foco em slicing
Hu et al. (2023)	DDPG-D3QN (ação híbrida)	Borda-nuvem, IoT	Latência e energia	Simulador não informado	Domínio de <i>offloading</i> , não RAN
Budhiraja et al. (2021)	DDDPG (DQN com DDPG) para PF	D2D subjacente	Sum-rate e fairness	Simulador não informado	Não trata eMBB/uRLLC em HetNet
Naderializadeh et al. (2021)	DRL multiagente descentralizado	HetNet densa	Taxa média e 5º percentil	Simulador não informado	Controla potência e usuário, não RB e QoS
Chang et al. (2022)	DRL multiagente (feedback de CQI)	5G NR <i>downlink</i>	Supera variantes de PF	Simulador não informado	Sem foco explícito em uRLLC e fairness
Hao et al. (2023)	RPPO sob POMDP (CSI parcial)	Multi-célula, multi-tráfego	Atraso de cauda e médio	Simulador não informado	Não modela HetNet multicamada eMBB/uRLLC

Trabalho	Técnica	Cenário / Tráfego	Objetivos (QoS / Fairness)	Avaliação	Limitação relevante a este trabalho
Kumbi et al. (2026)	DRL ciente de mobilidade e QoS	Rodovia, <i>downlink</i>	Eficiência energética e QoS	Simulador não informado	Cenário específico, sem <i>ensemble</i>
Pratap et al. (2021)	Algoritmos (NP-difícil) de <i>fairness</i>	HetNet de <i>small cells</i>	<i>Fairness</i> e reúso	Dados reais e simulação	Não emprega aprendizado
Stigenberg et al. (2021): QADRA	DRL ciente de QoS (fim a fim)	5G NR, fluxos com QoS	Satisfação de QoS e throughput	<i>Vienna 5G SLS</i>	Aloca RB diretamente, sem <i>fairness</i> explícito
Este trabalho: DRLQoSScheduler	DDPG (ação contínua) controlando heurístico de QoS, com <i>ensemble</i>	HetNet macro-pico-femto, eMBB/uRLLC	<i>Throughput</i> , <i>fairness</i> (Jain) e latência	<i>Vienna 5G SLS v1.4</i>	Proposta desta dissertação

Fonte: Elaborada pelo autor.

Nota: a coluna Avaliação identifica a ferramenta de simulação quando o trabalho a nomeia explicitamente. Emprega-se “simulação própria” quando os autores relatam implementação própria do ambiente, “simulação customizada” quando partem de plataforma existente com modificações, e “simulador não informado” quando o trabalho não especifica a ferramenta empregada.

A leitura transversal da literatura permite identificar tendências metodológicas claras. Observa-se, em primeiro lugar, a migração de escalonadores heurísticos de parâmetros fixos para soluções baseadas em aprendizado. Dentro do DRL, nota-se a evolução dos métodos baseados em valor, com ação discreta, rumo aos métodos de gradiente de política com ação contínua. Verifica-se, por fim, a crescente preocupação com o realismo, expressa no tratamento da observação parcial, da mobilidade e das restrições de tempo real. A despeito desses avanços, persistem lacunas de conhecimento que este trabalho se propõe a endereçar.

A primeira lacuna diz respeito ao controle indireto contínuo. A maioria dos escalonadores baseados em DRL aloca blocos de recurso de forma direta, e assim enfrenta um espaço de ação de crescimento combinatório, ou recorre à discretização, que sacrifica a granularidade. São escassos os trabalhos em que o agente atua como controlador de alto nível, ajustando de maneira contínua um pequeno conjunto de parâmetros de uma heurística de domínio já validada, estratégia que reduz a dimensão do problema, estabiliza o treinamento e preserva a interpretabilidade.

A segunda lacuna refere-se à *fairness* como objetivo controlável. A equidade costuma surgir como subproduto implícito da recompensa ou como objetivo de esquemas puramente

heurísticos. Falta uma formulação na qual o próprio agente module de forma explícita o nível de *fairness* desejado em resposta ao estado da rede, equilibrando eficiência e equidade de modo adaptativo.

A terceira lacuna envolve a integração de eMBB e uRLLC em HetNet multicamada. Os trabalhos de multiplexação eMBB/uRLLC concentram-se em células isoladas ou em modelos analíticos, ao passo que os estudos de HetNet raramente incorporam o tráfego misto com *deadlines* rígidos. A avaliação conjunta de tráfego eMBB/uRLLC sobre uma topologia macro-pico-femto, com escalonamento adaptativo por camada, ainda é pouco explorada.

A quarta lacuna trata da reprodutibilidade e da generalização. A fragmentação de simuladores compromete a comparação dos resultados, e a capacidade de generalização das políticas aprendidas, por exemplo o desempenho *zero-shot* em cenários distintos do de treinamento, é raramente avaliada de forma sistemática. Tampouco são comuns mecanismos, como o *ensemble* de agentes, voltados a mitigar a alta variância inerente ao treinamento por DRL.

É precisamente nessa interseção que se insere o DRLQoSScheduler proposto nesta dissertação. A solução emprega um agente DDPG de ação contínua para ajustar, a cada intervalo de decisão, três parâmetros de alto nível de um escalonador heurístico que toma a métrica PF como base e a pondera por fatores de QoS herdados do QAS e do CQAS, a saber, a urgência de latência, a penalidade de confiabilidade e o índice de *fairness* alvo. Dessa maneira, combinam-se as vantagens do controle contínuo, associadas à primeira lacuna, e da *fairness* explicitamente controlável, associada à segunda, avaliadas sobre uma HetNet macro-pico-femto com tráfego misto eMBB/uRLLC, o que responde à terceira lacuna, no simulador padronizado Vienna 5G SLS v1.4 e com análise de generalização e *ensemble* de agentes, o que atende à quarta lacuna. O detalhamento técnico dessa proposta é apresentado no Capítulo 5.

2.7 Considerações Finais

Este capítulo apresentou um panorama crítico da produção científica sobre o escalonamento de recursos em redes 5G, organizado em quatro eixos: as heurísticas orientadas a QoS e as suas limitações de parâmetros fixos; a multiplexação dos serviços eMBB e uRLLC e os seus fundamentos teóricos; a aplicação do aprendizado por reforço profundo ao RRM, das abordagens baseadas em valor às de controle contínuo e multiagente; e a equidade em redes heterogêneas. A síntese comparativa evidenciou que, embora o DRL já demonstre superioridade consistente sobre as heurísticas clássicas, permanecem abertas lacunas relativas

ao controle indireto contínuo, ao tratamento explícito da *fairness*, à integração de tráfego misto em HetNets multicamada e à reprodutibilidade e generalização das soluções. A proposta desta dissertação posiciona-se na convergência dessas lacunas.

3 REDES 5G E ESCALONAMENTO DE RECURSOS

3.1 Arquitetura e Casos de Uso do 5G

A quinta geração de redes móveis (5G), padronizada pelo 3GPP a partir do *Release 15* no contexto do IMT-2020 definido pela UIT-R, foi concebida para atender simultaneamente a três cenários de uso com requisitos mutuamente conflitantes: *eMBB*, *uRLLC* e *mMTC* (ERUNKULU et al., 2021). Os requisitos mínimos de desempenho estabelecidos pela ITU-R para cada classe são apresentados na Tabela 3.1:

Tabela 3.1- Requisitos mínimos de desempenho das classes de serviço 5G (ITU-R, 2017)

KPI	eMBB	uRLLC	mMTC
Taxa de pico (DL/UL)	20/10 Gbps	—	—
Taxa de dados por usuário	100/50 Mbps	—	—
Latência no plano de usuário	4 ms	1 ms	—
Confiabilidade	—	99,999%	—
Densidade de conexões	—	—	10 ⁶ dispositivos/km ²
Eficiência espectral de pico (DL/UL)	30/15 b/s/Hz	—	—
Mobilidade máxima	500 km/h	—	—

Fonte: Adaptada de ITU-R (2017).

Embora o 5G contemple também o serviço mMTC, este trabalho concentra-se na coexistência eMBB/uRLLC, em que os requisitos de *throughput* e latência geram os principais desafios de escalonamento.

Do ponto de vista arquitetural, o 5G é estruturado em duas camadas: a Rede de Acesso por Rádio de Nova Geração (NG-RAN, do inglês *Next-Generation Radio Access Network*) e o Núcleo 5G (5GC, do inglês *5G Core Network*). A NG-RAN é composta por estações base denominadas gNodeB (gNB), responsáveis pelas funções de camada física, controle de acesso ao meio (MAC, do inglês *Medium Access Control*), controle de enlace de rádio (RLC, do inglês *Radio Link Control*) e protocolo de convergência de dados de pacote (PDCP, do inglês *Packet Data Convergence Protocol*). A gNB conecta-se ao 5GC pela interface NG e a outras gNBs pela interface Xn, viabilizando coordenação inter-célula e *handover* sem interrupção de serviço (3GPP TS 23.501, 2020).

O 5GC adota uma arquitetura baseada em serviços (SBA, do inglês *Service-Based Architecture*), na qual as funções de rede (NFs, do inglês *Network Functions*) são expostas

como microsserviços acessíveis por APIs baseadas em HTTP/2 (3GPP TS 23.501, 2020). As principais NFs incluem:

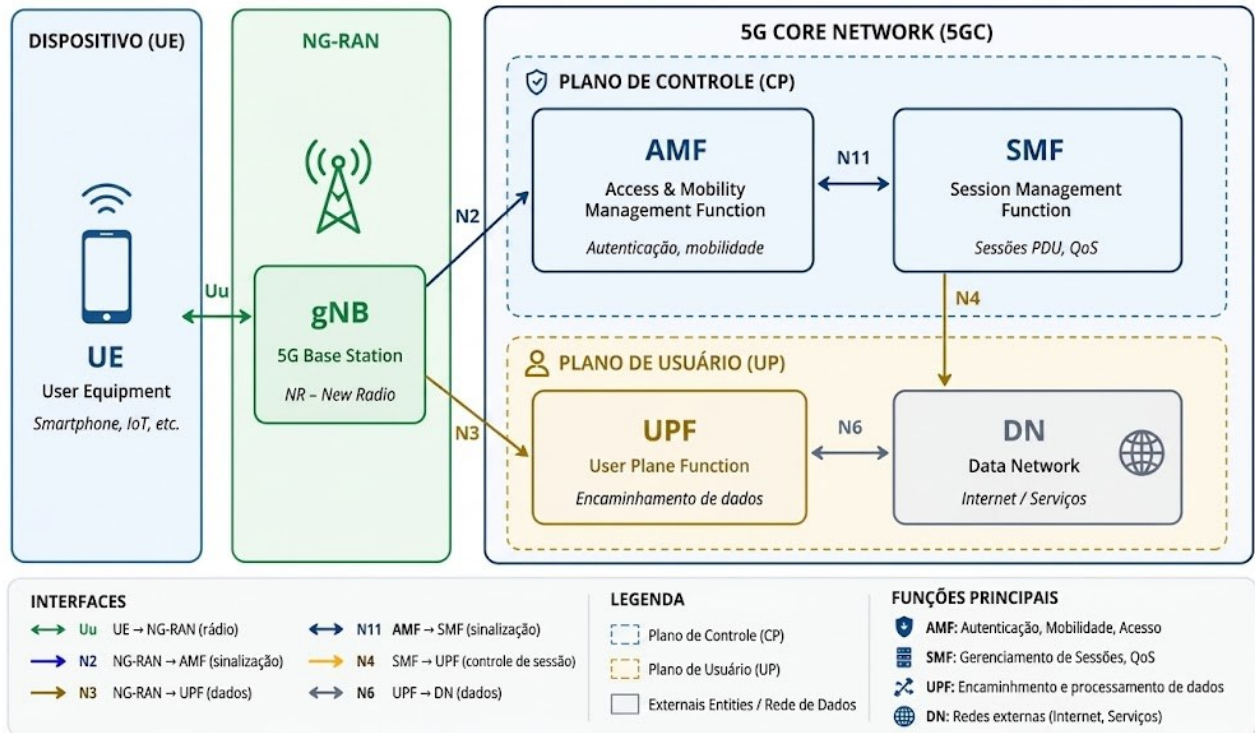
- *Access and Mobility Management Function* (AMF), responsável pelo registro, autenticação e mobilidade dos UEs;
- *Session Management Function* (SMF), que gerencia o estabelecimento e modificação de sessões PDU;
- *User Plane Function* (UPF), ponto de ancoragem do plano de dados entre a RAN e a rede de dados externa;
- *Policy Control Function* (PCF), que provê regras de QoS;
- *Network Slice Selection Function* (NSSF), que seleciona a fatia de rede adequada para cada sessão.

A separação entre plano de controle e plano de dados (CUPS, do inglês *Control and User Plane Separation*) permite escalabilidade independente e suporte nativo a *network slicing*, pelo qual uma mesma infraestrutura física instancia múltiplas redes lógicas isoladas com garantias de QoS distintas (DOGRA et al., 2021).

A Figura 3.1 ilustra a arquitetura da NG-RAN e do 5GC, com as interfaces NG (gNB–5GC) e Xn (gNB–gNB) e as principais funções da SBA.

A interface de rádio do 5G NR opera em duas faixas de frequência (FR, do inglês *Frequency Range*): FR1 (410 MHz – 7,125 GHz), adequada para implantações de longo alcance com bandas sub-6 GHz, e FR2 (24,25 GHz – 52,6 GHz), correspondente às ondas milimétricas (mmWave), com larguras de banda de canal de até 400 MHz por portadora (DOGRA et al., 2021). A camada física emprega multiplexação por divisão de frequência ortogonal com prefixo cíclico (CP-OFDM, do inglês *Cyclic Prefix Orthogonal Frequency Division Multiplexing*) no *downlink* e tanto CP-OFDM quanto DFT-s-OFDM no *uplink*, com espaçamentos entre subportadoras (numerologias μ) de $15 \cdot 2^\mu$ kHz, para $\mu \in \{0,1,2,3,4\}$, conforme especificado no 3GPP TS 38.211. O espaçamento de subportadora determina diretamente a duração do *slot*, a unidade mínima de escalonamento, variando de 1 ms ($\mu=0$) a 0,125 ms ($\mu=3$), permitindo que a camada física se adapte ao perfil de latência de cada classe de serviço (ERUNKULU et al., 2021). A numerologia $\mu=4$ ($\Delta f = 240$ kHz, *slot* de 0,0625 ms), incluída na Tabela 3.2, é restrita à transmissão de blocos de sinal de sincronização (SSB, do inglês *Synchronization Signal Blocks*) em mmWave e não é considerada nos cenários de escalonamento deste trabalho.

Figura 3.1 - Arquitetura da NG-RAN e 5GC



Fonte: Elaborada pelo autor com base em 3GPP TS 23.501 (2020).

A coexistência das classes eMBB e uRLLC numa mesma célula constitui um problema fundamental de gerenciamento de RRM: usuários eMBB demandam maximização de *throughput* de longa duração, enquanto usuários uRLLC necessitam de alocação prioritária e imediata de recursos com garantias rígidas de *deadline*. Esse conflito é exacerbado HetNets, nas quais múltiplas camadas de células (macro, pico e femto) operam simultaneamente com potências e coberturas distintas, ampliando a variabilidade espacial das condições de canal (PANWAR et al., 2016). O problema de escalonamento em cenários com tráfego misto eMBB/uRLLC é o foco central deste trabalho, e a solução proposta é detalhada no Capítulo 5.

3.2 Estrutura de Transmissão (NR, TTI, RB)

A interface de rádio do 5G NR foi concebida com uma estrutura de tempo-frequência flexível, capaz de suportar simultaneamente os requisitos distintos das classes eMBB, uRLLC e mMTC sobre uma mesma portadora. O princípio central dessa flexibilidade é a numerologia escalável, que parametriza a forma de onda OFDM por meio do espaçamento entre subportadoras $\Delta f = 2^\mu \cdot 15 \text{ kHz}$, onde $\mu \in \{0, 1, 2, 3, 4\}$, conforme definido no 3GPP TS 38.211 (3GPP TS 38.211, 2020).

3.2.1 Frame, Subframe e Slot

A estrutura temporal do NR é hierárquica: um quadro (*frame*) possui duração fixa de 10 ms e é subdividida em 10 subquadros (*subframes*) de 1 ms cada assim como no *Long Term Evolution* (LTE), preservando compatibilidade para coexistência entre as duas gerações. Cada subquadro contém 2^μ *slots*, e cada *slot* é composto por 14 símbolos OFDM (com prefixo cíclico normal) (LIN et al., 2019). A duração de um *slot* varia, portanto, de 1 ms ($\mu=0$, $\Delta f=15$ kHz) a 0,125 ms ($\mu=3$, $\Delta f=120$ kHz), como sintetizado na Tabela 3.2.

Tabela 3.2 - Parâmetros de numerologia do 5G NR (3GPP TS 38.211, 2020)

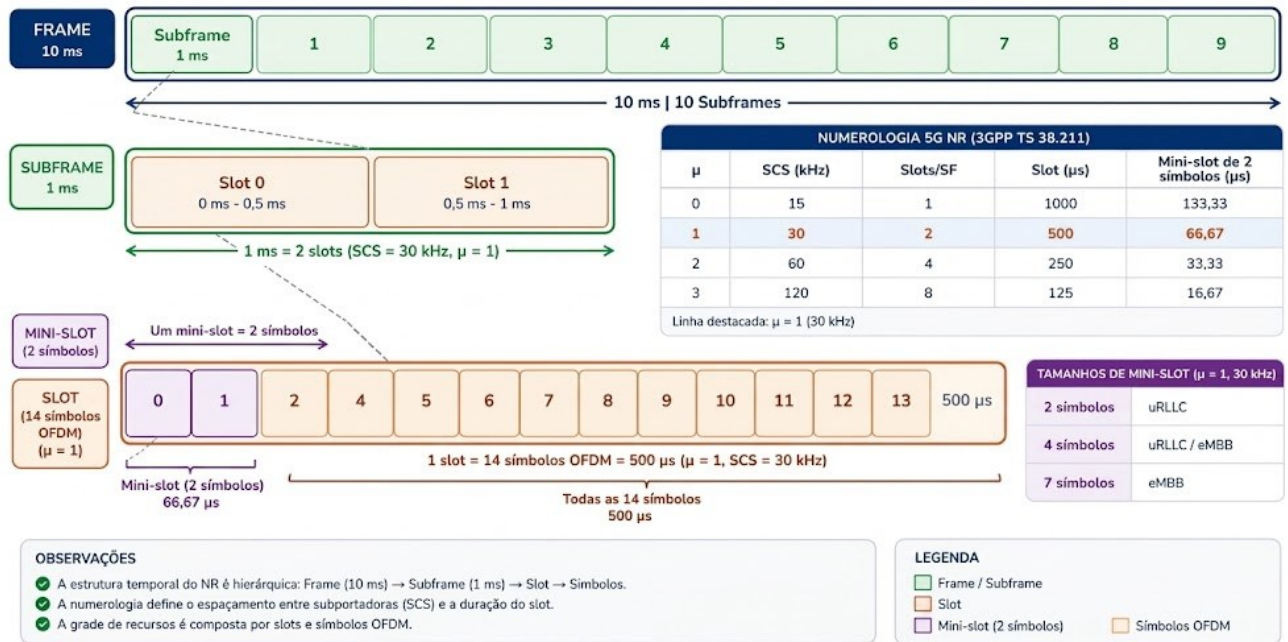
M	Δf (kHz)	Slots/subquadro	Duração do slot (ms)	Aplicação típica
0	15	1	1,000	eMBB sub-6 GHz
1	30	2	0,500	eMBB sub-6 GHz
2	60	4	0,250	eMBB/uRLLC sub-6 GHz
3	120	8	0,125	uRLLC/mmWave
4	240	16	0,0625	SSB mmWave

Fonte: Adaptada de 3GPP TS 38.211 (2020).

A Figura 3.2 sintetiza a hierarquia *frame–subframe–slot–mini-slot* do 5G NR para as numerologias $\mu \in \{0,1,2,3\}$, evidenciando a redução da duração do *slot* com o aumento de μ .

O *slot* é a unidade mínima de escalonamento no NR, porém o padrão introduz o conceito de *mini-slot*, que permite transmissões de 2, 4 ou 7 símbolos OFDM com início em qualquer símbolo do *slot* (MARIJANOVIĆ et al., 2021). Esse mecanismo é fundamental para o uRLLC, pois viabiliza preempção de transmissões eMBB em andamento: ao receber dados uRLLC urgentes, o escalonador pode interromper (*puncture*) os símbolos eMBB e alocar o *mini-slot* imediatamente, sem aguardar o próximo limite de *slot*, assim reduzindo a latência de acesso ao canal para menos de 0,5 ms (MARIJANOVIĆ et al., 2021).

Figura 3.2 - Estrutura temporal NR



Fonte: Elaborada pelo autor com base em 3GPP TS 38.211 (2020).

3.2.2 Grade de Recursos, Resource Block e Resource Element

No domínio da frequência, a portadora NR é subdividida em subportadoras indexadas por k . O agrupamento de 12 subportadoras consecutivas ao longo de um *slot* forma um bloco de recursos (RB, do inglês *Resource Block*), que constitui a unidade mínima de alocação de frequência pelo escalonador diferentemente do LTE, o RB no NR é definido exclusivamente no domínio da frequência, sem extensão temporal fixa (3GPP TS 38.211, 2020). Uma portadora NR do *Release 15* suporta até 3 300 subportadoras ativas (275 RBs), com largura de banda máxima de 100 MHz na FR1 e 400 MHz na FR2, o que é significativamente superior ao limite de 20 MHz do LTE (LIN et al., 2019).

A interseção de uma subportadora k com um símbolo OFDM l define um elemento de recursos (RE, do inglês *Resource Element*), que é a menor unidade endereçável da grade de recursos. Formalmente, um RE é identificado pelo par (k, l) na grade de uma parte de largura de banda (BWP, do inglês *Bandwidth Part*), um subconjunto contíguo de RBs configurado pela gNB para cada equipamento de usuário (UE, do inglês *User Equipment*). O NR permite até quatro BWPs configuradas por UE, com apenas uma ativa por direção de transmissão em cada instante, viabilizando *bandwidth adaptation*: UEs em modo de baixa atividade operam numa BWP estreita, comutando dinamicamente para uma BWP larga quando o escalonador aloca recursos adicionais (LIN et al., 2019).

3.2.3 Adaptação de Enlace e Indicador de Qualidade do Canal

O processo de alocação de recursos é indissociável da *Adaptação de Enlace* (AMC, do inglês *Adaptive Modulation and Coding*), pelo qual a gNB seleciona o esquema de modulação e codificação (MCS, do inglês *Modulation and Coding Scheme*) mais adequado às condições instantâneas do canal de cada UE. O NR suporta os esquemas de modulação QPSK, 16-QAM, 64-QAM e 256-QAM, com códigos de canal LDPC (*Low-Density Parity-Check*) para canais de dados e códigos polares para canais de controle (LIN et al., 2019).

A estimativa da qualidade do canal é realizada pelo UE e reportada à gNB na forma de indicador de CQI, indicador de matriz de pré-codificação (PMI, do inglês *Precoding Matrix Indicator*) e indicador de posto (RI, do inglês *Rank Indicator*), coletivamente denominados informação de estado de canal (CSI, do inglês *Channel State Information*). O CQI mapeia a SINR medida em um índice de 4 bits (0–15), que o escalonador utiliza para selecionar o MCS e estimar o *throughput* esperado por RB alocado, parâmetro central nos algoritmos de escalonamento baseados em eficiência espectral, como o *Proportional Fair* (PF), detalhado na Seção 3.7 (LIN et al., 2019).

A compreensão da estrutura de transmissão descrita nesta seção é indispensável para o projeto do escalonador proposto: o espaço de ação do agente DDPG opera sobre parâmetros que modulam a priorização de RBs a cada TTI (no contexto do NR, TTI corresponde ao *slot* ou mini-slot, conforme a numerologia utilizada), e a função de recompensa incorpora métricas diretamente derivadas da grade de recursos como *throughput*, BLER e violação de *deadline*, conforme detalhado no Capítulo 5.

3.3 Tráfego Misto: eMBB e uRLLC

A coexistência de serviços eMBB e uRLLC numa mesma célula 5G configura um dos problemas mais desafiadores de gerenciamento de recursos de rádio, pois impõe objetivos fundamentalmente conflitantes sobre a mesma grade de tempo-frequência: maximizar o *throughput* de longo prazo dos usuários eMBB e, simultaneamente, garantir latência e confiabilidade extremas para os usuários uRLLC (ANAND et al., 2020).

3.3.1 Modelos de Tráfego

Os dois serviços possuem características estatísticas de tráfego essencialmente distintas. O tráfego eMBB é modelado como *Full Buffer*: os usuários sempre têm dados a transmitir, de modo que o escalonador pode otimizar continuamente a eficiência espectral com base no CQI instantâneo, sem restrições de *deadline*. Trata-se de um modelo de tráfego adequado para aplicações de *streaming* de vídeo em alta definição, realidade virtual e acesso à internet de banda larga (ANAND et al., 2020).

O tráfego uRLLC, por contraste, é modelado como um processo de chegada de Poisson com pacotes curtos e esporádicos. Cada pacote possui um *deadline* rígido D , tipicamente entre 0,25 ms e 1 ms, e o PER (do inglês, Packet Error Rate) exigido situa-se na faixa de 1×10^{-5} a 1×10^{-9} , o requisito típico é $PER \leq 10^{-5}$ (confiabilidade de 99,999%, conforme Tabela 3.1), podendo ser tão restrito quanto 10^{-9} em aplicações industriais críticas (POPOVSKI et al., 2018). Ao contrário do eMBB, cujo desempenho é avaliado em termos de taxa média, o uRLLC é avaliado não pela taxa média, mas pelos eventos extremos da distribuição: a fração de pacotes que violam o *deadline* (*latency violation rate*) e a taxa de erro de bloco (BLER, do inglês *Block Error Rate*). A natureza intermitente e *bursty* do tráfego uRLLC implica que os recursos eventualmente reservados ficam ociosos na ausência de chegadas, gerando ineficiência espectral (POPOVSKI et al., 2018).

3.3.2 Estratégias de Multiplexação

Para acomodar os dois serviços, três abordagens principais têm sido estudadas na literatura (LIU et al., 2024):

- Reserva estática de recursos (*Resource Reservation*): uma fração fixa da grade de tempo-frequência é pré-alocada exclusivamente ao uRLLC, e o restante é destinado ao eMBB. Embora garanta os requisitos de latência e confiabilidade do uRLLC, essa estratégia desperdiça recursos durante os períodos sem chegadas uRLLC, degradando o *throughput* eMBB (LIU et al., 2024).
- Preempção e *Puncturing*: os recursos são alocados ao eMBB como padrão. Ao chegarem pacotes uRLLC urgentes, o escalonador interrompe (*punctures*) os símbolos eMBB em andamento e transmite o uRLLC imediatamente no próximo *mini-slot* sem aguardar o limite de *slot*. A gNB sinaliza ao UE eMBB as posições

pontuadas por meio do DCI *format 2_1*, formato de informação de controle de *downlink* definido pelo 3GPP como indicação de preempção, que informa ao UE quais recursos previamente a ele alocados foram interrompidos, conforme especificado no 3GPP TS 38.213, para que o UE possa reconstruir a transmissão com possível perda de taxa (ANAND et al., 2020; 3GPP TS 38.213, 2020). A perda de taxa eMBB é modelada como função do volume de *puncturing*: linear, convexa ou por limiar (*threshold*), com implicações distintas para a estratégia ótima de escalonamento conjunto (ANAND et al., 2020).

- Múltiplo Acesso Não-Ortogonal Heterogêneo (H-NOMA): os recursos são compartilhados de forma não-ortogonal entre eMBB e uRLLC, com transmissão simultânea e decodificação por cancelamento sucessivo de interferência (SIC, do inglês *Successive Interference Cancellation*). Embora possa superar a alocação ortogonal em regimes específicos de carga, o H-NOMA exige potência suficiente para garantir a confiabilidade uRLLC e introduz interferência mútua que pode comprometer ambos os serviços (POPOVSKI et al., 2018; LIU et al., 2024).

3.3.3 O Problema Multiobjetivo de Escalonamento

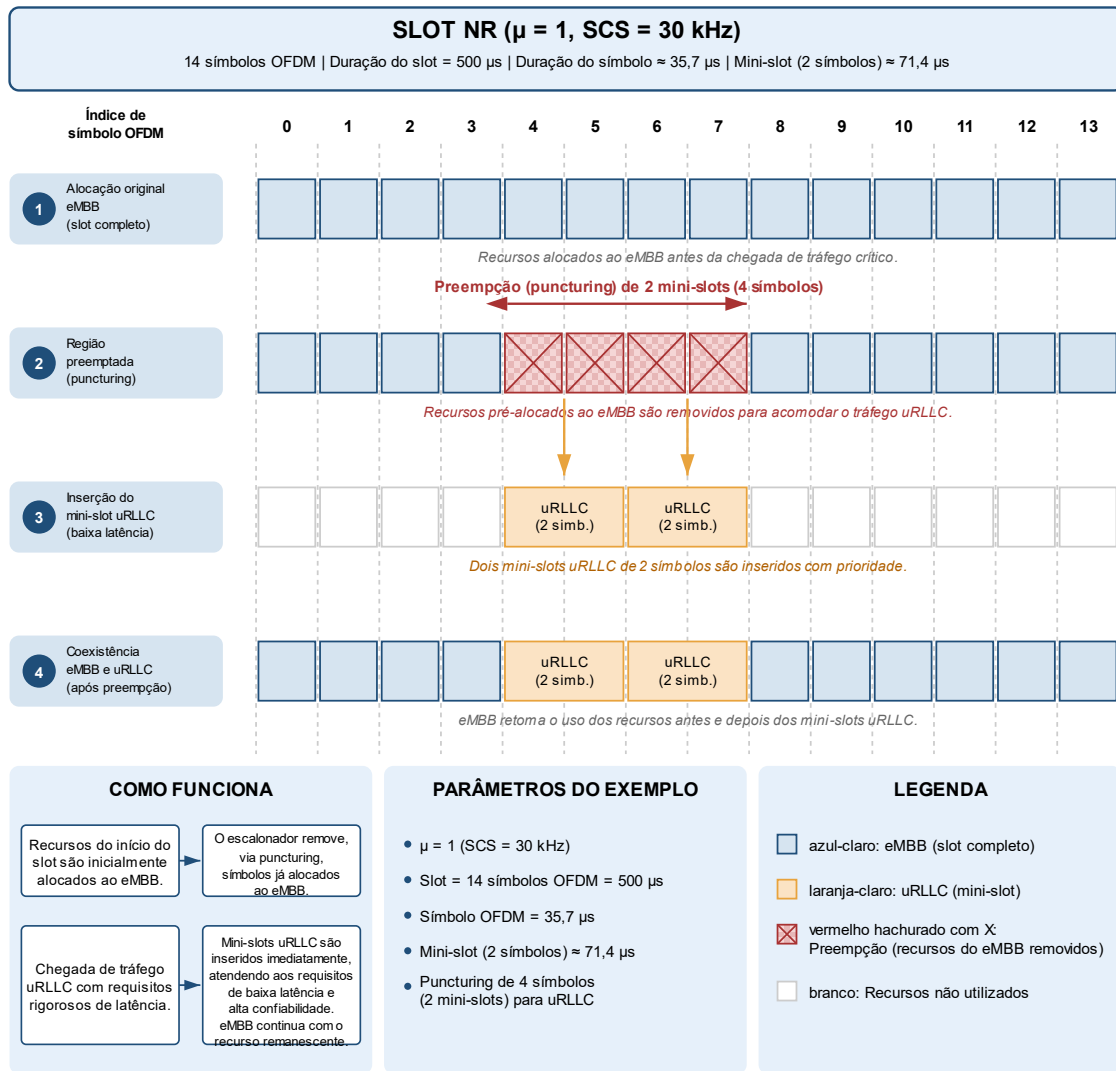
A coexistência eMBB/uRLLC define formalmente um problema de otimização multiobjetivo. Seja $\mathcal{U}_E$ o conjunto de usuários eMBB e $\mathcal{U}_U$ o conjunto de usuários uRLLC, o escalonador deve, a cada TTI, resolver:

$$\max_x \sum_{i \in \mathcal{U}_E} U_i(R_i(x)); \Pr[\text{deadline}_j \text{ violado}] \leq \epsilon_j, \quad \forall j \in \mathcal{U}_U \quad (3.1)$$

onde x é a matriz de alocação de RBs, U_i é a função utilidade do usuário eMBB _{i} , R_i é a taxa obtida, e $\epsilon_j \leq 10^{-5}$ é a probabilidade máxima de violação de *deadline* do usuário uRLLC _{j} (ANAND et al., 2020). A dificuldade fundamental desse problema reside na escala temporal: as decisões eMBB ocorrem no nível do *slot* (1 ms para $\mu=0$), enquanto os eventos uRLLC exigem resposta no nível do mini-slot (0,125–0,5 ms), criando um acoplamento entre dois horizontes de otimização distintos (ANAND et al., 2020).

A Figura 3.3 ilustra a coexistência dos fluxos eMBB e uRLLC na grade de recursos tempo-frequência, com destaque para o mecanismo de *puncturing* pelo qual transmissões uRLLC urgentes interrompem símbolos eMBB em andamento.

Figura 3.3 - Coexistência entre eMBB e uRLLC.



Fonte: Elaborada pelo autor com base em 3GPP TS 38.300 e TS 38.214 (2022).

Agravando esse cenário, a equidade entre usuários eMBB, aqui mensurada pelo índice de $J = (\sum_i R_i)^2 / (n \sum_i R_i^2)$ não é naturalmente preservada quando o escalonador prioriza uRLLC: usuários eMBB com melhores condições de canal tendem a absorver desproporcionalmente as interrupções por *puncturing*, gerando políticas enviesadas (LIU et al., 2024). A solução proposta neste trabalho aborda explicitamente esse aspecto por meio de um mecanismo de controle de *fairness* integrado ao agente DDPG, descrito no Capítulo 5.

3.4 Qualidade de Serviço (QoS)

No contexto das redes 5G, a qualidade de serviço (QoS, do inglês *Quality of Service*) designa o conjunto de mecanismos que garantem tratamento diferenciado ao tráfego de diferentes aplicações, assegurando que os requisitos de desempenho de cada serviço sejam atendidos de forma controlável e previsível. O *framework* de QoS do 5G foi reformulado em relação ao LTE, introduzindo o conceito de *QoS Flow* como unidade fundamental de tratamento de tráfego no plano de usuário (3GPP TS 23.501, 2020).

3.4.1 Framework de QoS do 5G

Cada sessão PDU entre um UE e a rede de dados pode conter múltiplos *QoS Flows*, cada um identificado por um *QoS Flow Identifier* (QFI). Um *QoS Flow* agrupa pacotes que recebem o mesmo tratamento de encaminhamento definido por parâmetros como peso de escalonamento, limiar de admissão e configuração de protocolo de enlace. A associação entre um *QoS Flow* e seus parâmetros de desempenho é realizada por meio do *5G QoS Identifier* (5QI), um escalar que referencia um conjunto padronizado de características de QoS (3GPP TS 23.501, 2020).

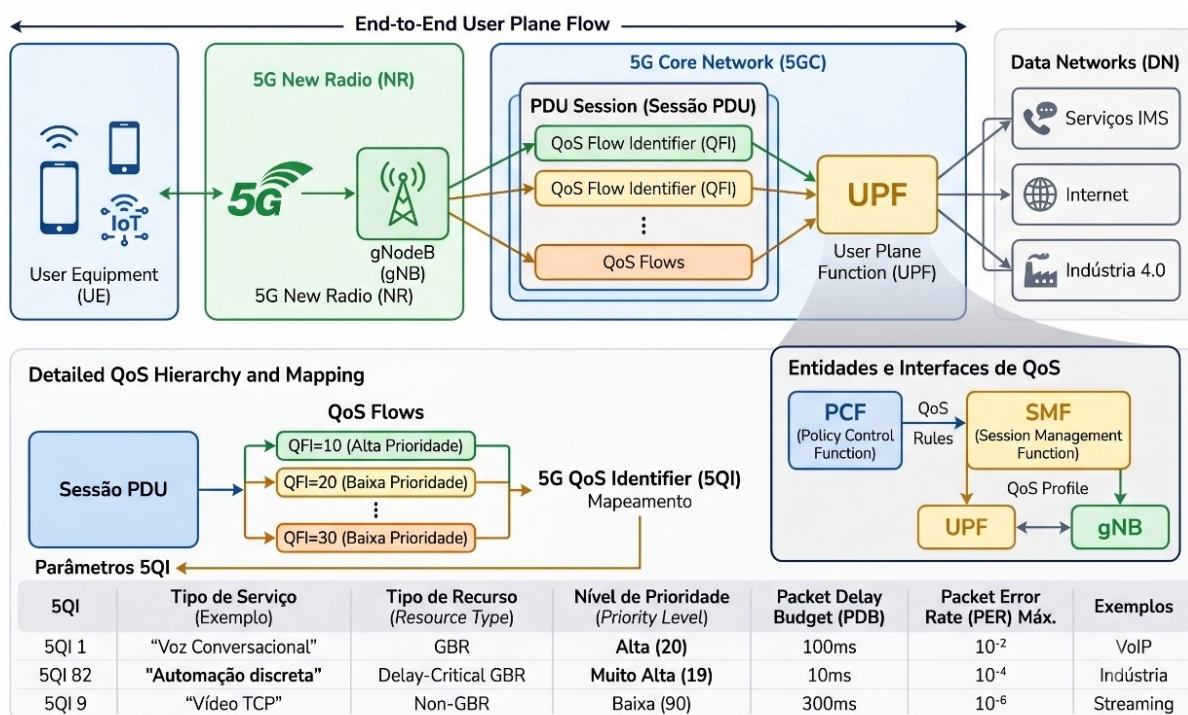
As características de QoS associadas a cada 5QI compreendem cinco parâmetros principais (3GPP TS 23.501, 2020; MAMANE et al., 2022):

- Tipo de Recurso (*Resource Type*): classifica o fluxo em GBR, *Delay-Critical GBR* ou Non-GBR. Fluxos GBR possuem taxa mínima garantida (*GFBR*) e taxa máxima (*MFBR*); fluxos Non-GBR compartilham recursos sem garantia de taxa mínima.
- Nível de Prioridade (*Priority Level*): escalar de 1 (mais alta) a 127, utilizado pelo escalonador para desempate na alocação de RBs.
- Orçamento de Atraso de Pacote (PDB, do inglês *Packet Delay Budget*): latência máxima tolerada entre o UE e a UPF, em milissegundos, que orienta o dimensionamento de filas e a prioridade de escalonamento.
- Taxa de Erro de Pacote (PER, do inglês *Packet Error Rate*): probabilidade máxima de perda ou erro de pacote não detectado pela camada de enlace, diretamente relacionada à confiabilidade exigida pelo serviço.

- Volume Máximo de Rajada de Dados (*Maximum Data Burst Volume*): parâmetro adicional para fluxos *Delay-Critical GBR*, que limita o volume de dados por rajada — relevante para serviços industriais e V2X.

A Figura 3.4 sintetiza o *framework* de QoS do 5G, relacionando o *QoS Flow*, identificado pelo QFI, ao respectivo 5QI e aos parâmetros de desempenho associados, o orçamento de atraso de pacote e a taxa máxima de erro de pacote.

Figura 3.4- Framework de QoS do 5G (QoS Flow/ 5QI/ PDB/ PER).



Fonte: Elaborada pelo autor com base em 3GPP TS 23.501 (2020).

Tabela 3.3 - Subconjunto de valores 5QI padronizados pelo 3GPP TS 23.501

5QI	Tipo de Serviço	Prioridade	Delay Budget	PER Máx.	GBR
1	Voz conversacional	20	100 ms	10^{-2}	Sim
4	Vídeo não conversacional (eMBB)	50	300 ms	10^{-6}	Sim
5	Sinalização IMS (eMBB)	10	100 ms	10^{-6}	Não
9	Vídeo buffered / TCP (eMBB)	90	300 ms	10^{-6}	Não
82	Automação discreta (uRLLC)	19	10 ms	10^{-4}	Sim
85	V2X crítico (uRLLC)	21	5 ms	10^{-5}	Sim

Fonte: Adaptada de 3GPP TS 23.501 (2020).

A Tabela 3.3 apresenta um subconjunto dos valores 5QI padronizados pelo 3GPP, ilustrando a diversidade de requisitos entre classes de serviço.

3.4.2 Fatores de Escalonamento Orientados a QoS

O algoritmo de escalonamento presente na gNB utiliza os parâmetros 5QI como entradas diretas no processo de decisão de alocação de recursos (MAMANE et al., 2022). Os principais fatores considerados são:

- **CQI e Condições de Canal:** o reportado pelo UE determina o MCS aplicável e, consequentemente, o número de bits por RE. Um CQI elevado viabiliza esquemas de alta ordem (256-QAM), maximizando a eficiência espectral por RB alocado.
- **Atraso *Head-of-Line* (HOL):** tempo que o pacote mais antigo da fila aguarda para ser transmitido, calculado como $D_{HOL} = t - t_{enter}$, onde t_{enter} é o instante de chegada ao *buffer* da gNB. Esse parâmetro é crítico para serviços uRLLC, cujo *deadline* pode ser violado se o HOL exceder o PDB.
- **Status de *Buffer* e Filas:** o relatório de estado de *buffer* (BSR, do inglês *Buffer Status Report*) enviado pelo UE informa o volume de dados pendente, permitindo que o escalonador priorize usuários com filas críticas e reduza a latência de entrega.
- **5QI e Requisitos de Confiabilidade:** o escalonador utiliza o 5QI para determinar o nível de prioridade, o PDB e a PER máxima tolerada, condicionando o número de retransmissões do protocolo de solicitação de repetição automática híbrida (HARQ, do inglês *Hybrid Automatic Repeat Request*) e a agressividade do MCS selecionado. Serviços com $PER \leq 10^{-5}$, como uRLLC, requerem MCS conservadores e potencial repetição de pacotes para garantir confiabilidade.

3.4.3 Métricas de Avaliação de QoS

A avaliação do desempenho de escalonadores em cenários com tráfego misto é realizada por um conjunto de métricas complementares (MAMANE et al., 2022):

- *Throughput* médio: taxa de bits recebidos corretamente por unidade de tempo, medida em Mbps, reflete a eficiência espectral global do sistema.
- Taxa de violação de latência (*Latency Violation Rate*): fração de pacotes uRLLC cujo *deadline* é excedido, a principal métrica de conformidade QoS para o uRLLC.
- BLER (*Block Error Rate*): fração de blocos de transporte com erro após HARQ, indicador direto de confiabilidade de enlace.
- Índice de Jain (*Jain's Fairness Index*): $J = (\sum_i R_i)^2 / (n \sum_i R_i^2) \in [1/n, 1]$ mede a equidade na distribuição de recursos entre usuários eMBB; $J=1$ indica distribuição perfeitamente equitativa.
- Percentil 5 do *throughput*: taxa obtida pelos 5% de usuários com pior desempenho, indicador de cobertura de usuários em bordas de célula.

Essas métricas são adotadas diretamente na função de recompensa do agente DDPG proposto neste trabalho (Capítulo 5) e na avaliação experimental do Capítulo 6.

3.5 Redes Heterogêneas (HetNets)

As Redes Heterogêneas (HetNets, do inglês *Heterogeneous Networks*) constituem uma das principais estratégias de densificação adotadas pelo 5G para atender ao crescimento exponencial do tráfego de dados. O princípio central é a sobreposição de nós de baixa potência (LPNs, do inglês *Low Power Nodes*) como nas picocélulas, femtocélulas e microcélulas sobrepostas à cobertura das macrocélulas existentes, formando uma arquitetura multicamada (*multi-tier*) capaz de combinar ampla cobertura com alta capacidade espectral por área (AGARWAL et al., 2022).

3.5.1 Arquitetura Multicamada

A arquitetura HetNet é caracterizada pela coexistência de nós com características heterogêneas de potência de transmissão, raio de cobertura e capacidade de *backhaul*. A Tabela 3.4 sintetiza os principais parâmetros de cada camada.

Os principais objetivos das HetNets são: (i) densificação celular para aumento de capacidade por área; (ii) aproximação das estações base aos usuários, reduzindo a perda de percurso e melhorando a eficiência espectral; (iii) absorção de tráfego (*offloading*) das

macrocélulas para células menores em regiões de alta densidade; e (iv) eliminação de zonas de sombra (*coverage holes*), garantindo uma experiência uniforme ao usuário (AGARWAL et al., 2022; ANAND; MUNTEAN, 2026).

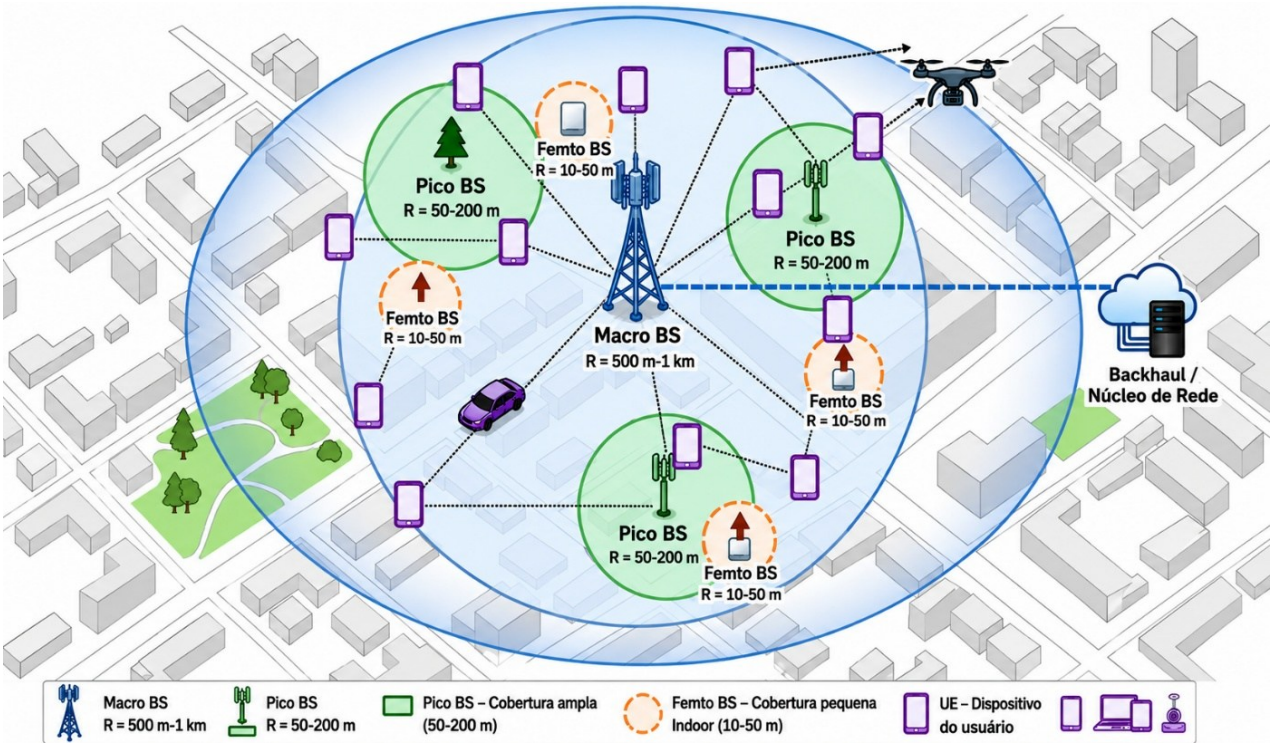
Tabela 3.4 - Características das camadas na arquitetura HetNet

Tipo de Célula	Potência Tx (dBm)	Raio de Cobertura	Localização Típica
Macrocélula	46	0,5 -5 km	Torres externas, cobertura ampla
Microcélula	30-37	100 -500 m	Áreas urbanas densas
Picocélula	23-30	10 -200 m	Hotspots, ambientes internos/externos
Femtocélula	≤ 23	10 -50 m	Residências, escritórios

Fonte: Elaborada pelo autor.

A Figura 3.5 exemplifica uma topologia HetNet típica, com macrocélula, picocélula e femtocélula coexistindo na mesma área de cobertura e compartilhando o espectro de frequência.

Figura 3.5 - Arquitetura HetNet (Rede Heterogênea de 3 camadas)



Fonte: Elaborada pelo autor com base em 3GPP TR 36.814 (2010).

3.5.2 Modos de Acesso e Associação de Usuários

As femtocélulas e picocélulas operam em dois modos de acesso principais (AGARWAL et al., 2022):

- Acesso Aberto (OSG, do inglês *Open Subscriber Group*): qualquer UE pode se associar ao nó de baixa potência. Favorece o *offloading* de tráfego, mas pode aumentar a interferência entre camadas.
- Acesso Fechado (CSG, do inglês *Closed Subscriber Group*): apenas UEs autorizados podem acessar o nó. Garante recursos aos assinantes cadastrados, porém UEs não autorizados nas proximidades sofrem interferência sem poder se associar — fenômeno denominado *victim UE*.

A regra de associação mais comum é baseada na máxima potência de sinal recebido (RSRP, do inglês *Reference Signal Received Power*), porém essa abordagem tende a sobrecarregar as macrocélulas enquanto as picocélulas ficam subutilizadas. Para corrigir esse desequilíbrio, o 3GPP introduziu a expansão de alcance de célula (CRE, do inglês *Cell Range Expansion*), que adiciona um viés (*bias factor*) B ao RSRP das células menores, incentivando UEs a migrarem para picocélulas mesmo com SINR mais baixa às custas de maior exposição à interferência inter-camadas (GULIA et al., 2022).

3.5.3 Interferência em HetNets

A principal desvantagem das HetNets é o agravamento da interferência intercelular, que ocorre em duas modalidades (AGARWAL et al., 2022; SIDDIQUI et al., 2021):

- Interferência entre camadas (*Cross-tier interference*): ocorre quando macrocélulas e células menores compartilham os mesmos RBs no domínio da frequência. Um UE associado a uma femtocélula pode receber interferência dominante da macrocélula vizinha no *downlink* e vice-versa. A assimetria de potência entre as camadas torna esse tipo de interferência especialmente severo para UEs na borda de cobertura das células menores.
- Interferência intra-camada (*Co-tier interference*): ocorre entre células da mesma camada quando há reuso de frequência agressivo em implantações densas. Femtocélulas adjacentes compartilhando os mesmos RBs se interferem mutuamente, degradando a SINR dos usuários servidos.

Para mitigar esses efeitos, as principais técnicas padronizadas pelo 3GPP incluem (ALZUBAIDI et al., 2025; GULIA et al., 2022):

- ICIC/eICIC (*Inter-Cell Interference Coordination*): coordenação de recursos entre células vizinhas via interface X2, com técnicas de tempo (ABS, do inglês *Almost Blank Subframes*) e frequência (FFR, do inglês *Fractional Frequency Reuse*) para reduzir colisões de recursos.
- CoMP (*Coordinated MultiPoint*): múltiplos pontos de transmissão coordenam suas transmissões para servir cooperativamente UEs na borda de célula, convertendo interferência em sinal útil por meio de *Joint Transmission* (JT-CoMP) ou escalonamento coordenado.
- Controle de potência (*Power Control*): ajuste dinâmico da potência de transmissão de cada nó para limitar o raio de interferência, especialmente em femtocélulas de acesso fechado.

3.5.4 RRM em HetNets e Relevância para Este Trabalho

O gerenciamento de recursos de rádio em HetNets envolve a otimização conjunta de três subproblemas interdependentes: associação de usuários (UA, do inglês *User Association*), alocação de recursos (RA, do inglês *Resource Allocation*) e controle de potência (PA, do inglês *Power Allocation*) (AGARWAL et al., 2022). A complexidade desse problema é agravada pela heterogeneidade dos modelos de tráfego de usuários eMBB com demanda contínua de banda e usuários uRLLC com chegadas esporádicas e *deadlines* rígidos que impõe requisitos assimétricos ao escalonador de cada camada.

No cenário heterogêneo avaliado neste trabalho (Capítulo 6), a infraestrutura é composta por macrocélulas, picocélulas e femtocélulas operando segundo o modelo de propagação *Urban Macro* 3GPP TR 38.901 (2020), com reuso de frequência universal entre camadas. A ausência de mecanismos de coordenação inter-camada explícitos (adotada como limitação computacional documentada) implica que a interferência entre camadas é tratada estatisticamente pelo simulador, e o agente DDPG opera sobre as métricas agregadas resultantes dessa interferência, sem conhecimento direto de sua origem.

3.6 Escalonamento de Pacotes em *Downlink*

A função de escalonamento de pacotes em *downlink*, executada pela gNB a cada TTI, determina quais usuários receberão quais RBs e com qual esquema de modulação e codificação, com base nas condições de canal, no estado das filas e nos requisitos de QoS de cada fluxo. Trata-se de um problema de otimização combinatorial NP-difícil na sua forma geral por redução ao problema *Multiple Knapsack*, demonstrada formalmente em trabalhos de alocação de RBs (MAMANE et al., 2022), razão pela qual soluções heurísticas e baseadas em aprendizado de máquina têm sido amplamente estudadas.

3.6.1 Processo de Escalonamento

O processo de escalonamento opera de forma iterativa a cada TTI, seguindo três etapas fundamentais (MAMANE et al., 2022; SHAHID et al., 2026): (i) Coleta de métricas: a gNB recebe os relatórios CSI (CQI, PMI, RI) dos UEs, os relatórios de estado de buffer BSRs e o atraso HOL de cada fila; (ii) Cálculo do *score*, para cada par usuário–RB (i, j), calcula-se uma métrica de prioridade $w(i, j)$ que combina condições de canal e requisitos de QoS; (iii) Alocação por argmax: cada RB j é atribuído ao usuário $i = \text{argmax}_i w(i, j)$, e o MCS é selecionado com base no CQI reportado.

A escolha da métrica $w(i, j)$ define o comportamento do escalonador e determina o compromisso entre eficiência espectral, equidade e conformidade com requisitos de latência. Os principais fatores que influenciam essa métrica são: o CQI instantâneo, que determina o *throughput* estimado por RB; o atraso HOL, que quantifica a urgência temporal dos pacotes em fila; o *Buffer Status Report*, que informa o volume de dados pendente; e o 5QI, que define o nível de prioridade, o PDB e a PER máxima do serviço (MAMANE et al., 2022).

3.6.2 Limitações em Cenários com Tráfego Misto

Os algoritmos de escalonamento clássicos apresentam limitações estruturais em cenários com tráfego misto eMBB/uRLLC (PRASAD; SUNNY, 2024; SHAHID et al., 2026): (i) Parâmetros fixos como os pesos de urgência e a janela de suavização são configurados estaticamente, sem adaptação a variações dinâmicas de carga ou mobilidade; (ii) Equidade versus latência, escalonadores que priorizam uRLLC tendem a privar usuários eMBB de recursos (*starvation*), enquanto aqueles que preservam equidade global violam *deadlines*

uRLLC sob alta carga; (iii) Sem controle explícito de *fairness*: o índice de Jain não é incorporado diretamente como restrição ou objetivo; (iv) Escalabilidade em HetNets pela decisão de escalonamento em ambientes multicamada aumenta o espaço de busca de forma combinatorial, tornando inviável a otimização exata a cada TTI (KUMBI et al., 2026).

Essas limitações motivam a abordagem proposta neste trabalho, na qual um agente DDPG ajusta dinamicamente os parâmetros do escalonador heurístico em função do estado observado da rede, conforme detalhado no Capítulo 5.

3.7 Algoritmos de Referência (RR, *Best CQI*, QoS-Aware, PF)

Esta seção descreve os algoritmos de escalonamento de referência da literatura. Destes, o *Round Robin*, o *Best CQI*, o QoS-Aware (QAS) e o *Proportional Fair* são utilizados como linha de base nas avaliações experimentais (Capítulo 6); o M-LWDF e o EXP/PF são apresentados para contextualização. Esses algoritmos representam diferentes estratégias de compromisso entre eficiência espectral, equidade e conformidade com requisitos de QoS, e são amplamente adotados como referência na literatura de redes 5G (MAMANE et al., 2022; QUEIROZ; SILVA, 2024). A Tabela 3.5 organiza esses algoritmos em quatro classes, conforme o critério de decisão que cada um privilegia: equidade, qualidade de canal, atraso ou requisitos de QoS.

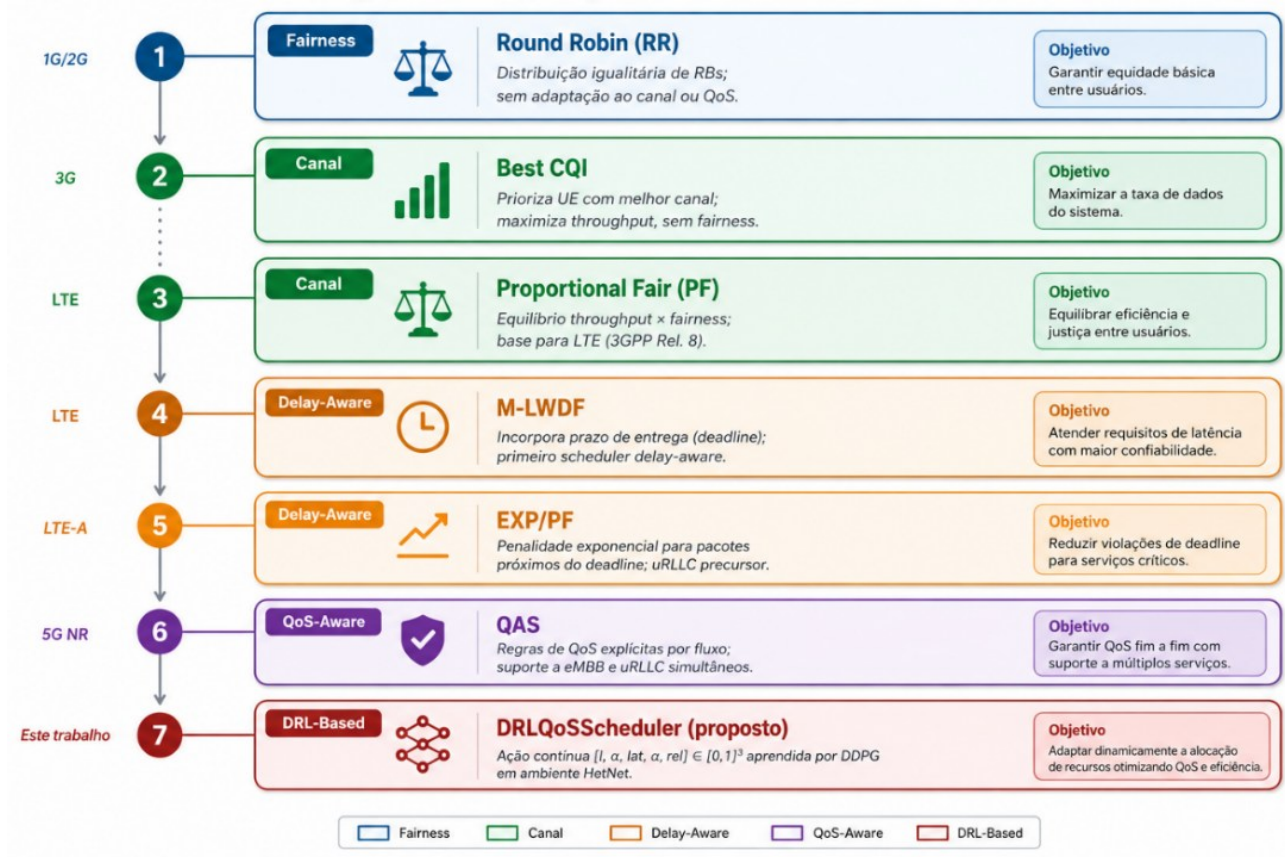
Tabela 3.5 - Classificação dos algoritmos de escalonamento em cenários 5G

Classe	Algoritmos
<i>Fairness</i>	RR, PF
<i>Channel-Aware</i>	<i>Best CQI</i>
<i>Delay-Aware</i>	M-LWDF, EXP/PF
<i>QoS-Aware</i>	QAS, CQAS

Fonte: Elaborada pelo autor.

A Figura 3.6 apresenta a evolução dos algoritmos de escalonamento analisados neste capítulo, do *Round Robin* ao DRLQoSScheduler proposto, classificados por critério de priorização.

Figura 3.6 - Evolução dos escalonadores de Downlink



Fonte: Elaborada pelo autor com base em 3GPP TS 36.213 (2022).

3.7.1 Round Robin (RR)

O *Round Robin* (RR) é o algoritmo de escalonamento mais simples, distribuindo os RBs ciclicamente entre todos os usuários ativos de forma equitativa em termos de tempo de canal. A métrica do RR é definida como:

$$m(i, k)^{RR} = t - T_i, \quad (3.2)$$

onde t representa o instante atual e T_i é o último instante em que o usuário i foi servido (QUEIROZ; SILVA, 2024). O RR garante igualdade de acesso ao canal, mas é insensível às condições de canal e aos requisitos de QoS, resultando em baixa eficiência espectral e incapacidade de atender *deadlines* uRLLC sob alta carga (MAMANE et al., 2022).

3.7.2 Best CQI

O escalonador *Best CQI* prioriza o usuário com melhor condição de canal em cada RB, maximizando a eficiência espectral instantânea da célula. A métrica é dada por:

$$m(i, k)^{Best-CQI} = \varsigma_i(\tau), \quad (3.3)$$

onde ς_i representa o valor de CQI do usuário i no instante τ . Embora maximize o *throughput* agregado da célula, o *Best CQI* penaliza severamente usuários em bordas de célula com baixo CQI, gerando distribuição altamente desigual de recursos (índice de Jain próximo de $1/n$) (MAMANE et al., 2022; ALQWIDER et al., 2022).

3.7.3 Proportional Fair (PF)

O *Proportional Fair* (PF) combina eficiência espectral com equidade histórica, priorizando usuários com boa condição de canal relativa à sua taxa média. A métrica PF é definida como:

$$w(i, j)^{PF} = \hat{T}_{i,j} / \bar{T}_i, \quad (3.4)$$

onde $\hat{T}_{i,j}$ é o *throughput* instantâneo estimado do usuário i no RB j , e $\bar{T}_i$ é a taxa média histórica do usuário i , atualizada a cada TTI como:

$$T_{bar\,i}(t) = (1 - 1/t_c) \times T_{bar\,i}(t - 1) + (1/t_c) \times T_{hat\,i}(t), \quad (3.5)$$

com t_c sendo a janela temporal de suavização. O PF maximiza o logaritmo da utilidade agregada, garantindo que nenhum usuário seja completamente privado de recursos. Contudo, por não incorporar informações de *deadline* ou confiabilidade, o PF puro é inadequado para tráfego uRLLC (MAMANE et al., 2022; SHAHID et al., 2026).

3.7.4 M-LWDF e EXP/PF

O *Modified Largest Weighted Delay First* (M-LWDF) estende o PF incorporando informação de atraso e confiabilidade para tráfego em tempo real. Sua métrica é:

$$w(i, j)^{MLWDF} = -[\log(\delta_i)/\tau_i] \times D_i^{HOL} \times [\hat{T}_{i,j}/\bar{T}_i], \quad (3.6)$$

onde δ_i é a probabilidade máxima de violação de *deadline*, τ_i é o *deadline* do pacote e D_i^{HOL} é o atraso *Head-of-Line*. O M-LWDF prioriza usuários com pacotes próximos da expiração e alta sensibilidade a atrasos (MAMANE et al., 2022).

O *Exponential Proportional Fair* (EXP/PF) combina o PF com um fator exponencial que amplifica a prioridade de usuários com HOL elevado em relação à média:

$$w(i, j)^{EXP/PF} = [T_{\text{hat } i}(i, j)/T_{\text{bar } i}(i)] \times e^{\left[\left(a_i \times D_i^{HOL} - \overline{aD^{HOL}} \right) / \left(1 + \left(\sqrt{\overline{aD^{HOL}}} \right) \right) \right]}, \quad (3.7)$$

onde $a_i = -\log(\delta_i)/\tau_i$ é o fator de urgência. O EXP/PF ativa a amplificação exponencial somente quando o atraso de algum usuário se desvia significativamente da média, mantendo o comportamento PF em condições normais (MAMANE et al., 2022).

3.7.5 QoS-Aware Scheduler (QAS/CQAS)

O *QoS-Aware Scheduler* (QAS) foi proposto originalmente por Shiyahin, Schwarz e Rupp (2022) para redes 5G com tráfego misto RT/NRT, utilizando o simulador Vienna 5G SLS. Posteriormente, Queiroz e Silva (2024) estenderam o QAS propondo o *Channel and QoS-Aware Scheduler* (CQAS), que integra adicionalmente as condições de canal (CQI) ao processo de priorização. O QAS é utilizado diretamente como linha de base nas avaliações experimentais desta dissertação; o CQAS é apresentado para contextualização, mas não foi incluído como linha de base, conforme detalhado ao final desta seção.

O QAS formula a alocação de RBs como um problema de maximização ponderada de *throughput*, com parâmetros de ajuste de latência e confiabilidade específicos de cada serviço (SHIYAHIN et al., 2022). Para cada *slot*, o problema de programação inteira binária é definido como:

$$\operatorname{argmax}_{b_1, \dots, b_i, c} c + \sum_i (\zeta_i \times t_i^T \times b_i), \quad (3.8)$$

sujeito a: $b(n)$ em $0,1$, $b_j^T \times b_k = 0$ para $k \neq j$, $t_i^T \times b_i \geq c \times \gamma_i$ para i em usuários *non-full-buffer*

onde $b_{n,k}$ é a variável binária de alocação do RB n ao usuário k , $t_{n,k}$ é o *throughput* estimado em bits, γ_k é o número de bits no *buffer* do usuário k , e c é uma variável de relaxação que garante viabilidade quando os RBs disponíveis são insuficientes. O parâmetro de ajuste $\zeta_k = \sigma^{-\max(d_{c,k} - d_k, 0)} \times \alpha^{-\beta_k}$ combina dois fatores: o fator de latência $\sigma = 1,05$ elevado ao complemento do atraso residual $(d_{c,k} - d_k)$, que prioriza usuários RT com pacotes próximos do *deadline*; e o fator de confiabilidade $\alpha = 2$ elevado ao negativo do BLER médio β_k , que prioriza usuários com baixa qualidade de enlace. Para usuários NRT, o expoente de latência é zero e, para $\beta_k > 0,1$, o fator de confiabilidade é neutralizado (SHIYAHIN et al., 2022). O DRLQoS Scheduler proposto nesta dissertação adapta a função de urgência de latência como $u_{lat}(i) = \sigma_{eff}^{(d(i)/d_c(i))}$, com $\sigma_{eff} = 1 + \alpha_{lat} \times (\sigma_{base} - 1)$ e $\alpha_{lat} \in [0,1]$ controlado pelo agente DRL; essa formulação difere do expoente negativo do atraso residual do QAS original e é detalhada no Capítulo 5.

O CQAS integra adicionalmente o CQI ao parâmetro de ajuste do QAS. Os valores de CQI são mapeados para o intervalo $I = [1,1; 1,15]$, e o parâmetro de ajuste final é obtido como o produto do fator de latência ($\sigma = 1,05$), do fator de confiabilidade ($\alpha = 2$) e do CQI normalizado. O *throughput* ajustado é então calculado como:

$$T_{ajustado}(i,j) = \zeta_i^C QAS \times \hat{T}_{i,j}, \quad (3.9)$$

sendo $\zeta_i^C QAS = \sigma \times \alpha \times CQI_{i_{norm}}$. Essa integração permite que o CQAS priorize simultaneamente usuários com requisitos críticos de QoS e boas condições de canal, obtendo ganhos de *throughput* de até 16,3% sobre o QAS e 163,6% sobre o RR para usuários de vídeo em cenários HetNet (QUEIROZ; SILVA, 2024).

O QoS-Aware fixo (QAS) (SHIYAHIN et al., 2022) é utilizado como referência principal nas avaliações dos Capítulo 6, por garantir comparabilidade direta e reprodutibilidade com os resultados de Shiyahin et al. (2022) e Queiroz e Silva (2024), que utilizaram o mesmo simulador e cenários equivalentes.

3.8 Considerações Finais

Este capítulo apresentou os fundamentos necessários para a compreensão do problema abordado neste trabalho. Foram introduzidas a arquitetura e os casos de uso do 5G NR, com ênfase nas classes eMBB e uRLLC e nos seus requisitos conflitantes de *throughput*, latência e confiabilidade. A estrutura de transmissão foi apresentada, incluindo o conceito de numerologia escalável, *slot*, *mini-slot*, *Resource Block* e *Bandwidth Part*, elementos que fornecem o substrato técnico sobre o qual o escalonador irá operar.

O problema de tráfego misto foi formalizado como uma otimização multiobjetivo, evidenciando as dificuldades inerentes à coexistência de serviços com horizontes temporais e requisitos qualitativos distintos. O *framework* de QoS do 5G, com seus parâmetros 5QI, PDB e PER, estabelece as métricas de conformidade que guiam tanto o escalonador heurístico quanto a função de recompensa do agente DRL.

A revisão das redes heterogêneas contextualizou o cenário de avaliação desta dissertação, em que macro, pico e femtocélulas coexistem com tráfego misto. Os algoritmos de referência (RR, *Best CQI*, PF, M-LWDF, EXP/PF e QoS-Aware) foram apresentados com suas métricas formais, evidenciando as limitações de abordagens com parâmetros fixos em ambientes dinâmicos.

4 APRENDIZADO POR REFORÇO PROFUNDO

O DRL integra os métodos de aprendizado por reforço (AR) com redes neurais profundas como aproximadores de função, permitindo que agentes aprendam políticas de decisão diretamente a partir de observações de alta dimensão em ambientes complexos e não-estacionários (SUTTON; BARTO, 2018; LI, 2018). Este capítulo apresenta as bases teóricas do AR e as técnicas específicas empregadas no escalonador proposto: o Processo de Decisão de Markov, os algoritmos *Q-Learning* e DQN como precursores, o Gradiente de Política Determinístico, o *Deep Deterministic Policy Gradient* (DDPG) e suas técnicas de estabilização, funções de recompensa multiobjetivo e a estratégia de *ensemble* de agentes.

4.1 Fundamentos de Aprendizado por Reforço

O AR estuda como um agente aprende a tomar sequências de decisões para maximizar um objetivo cumulativo por meio da interação com um ambiente desconhecido. Ao contrário do aprendizado supervisionado, que requer exemplos rotulados, o AR opera apenas com sinais de recompensa escalares, potencialmente esparsos e atrasados, tornando-o adequado para problemas em que o ambiente é demasiado complexo para ser modelado analiticamente (SUTTON; BARTO, 2018). Segundo Kaelbling, Littman e Moore (1996), o AR pode ser visto como uma generalização da programação dinâmica para o caso em que o modelo do ambiente é desconhecido.

4.1.1 Processo de Decisão de Markov

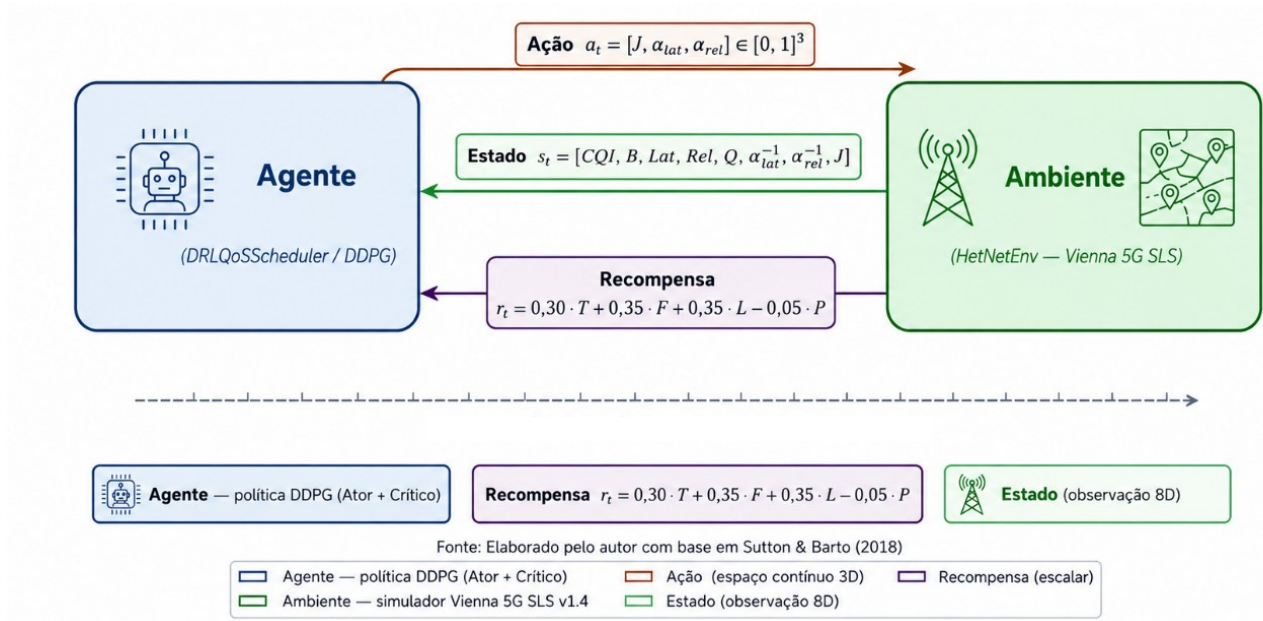
A formalização matemática do AR é dada pelo Processo de Decisão de Markov (MDP, do inglês *Markov Decision Process*), definido pela tupla (S, A, P, R, γ) , onde (SUTTON; BARTO, 2018): S é o espaço de estados, que pode ser discreto, contínuo ou híbrido; A é o espaço de ações, igualmente discreto ou contínuo; $P: S \times A \times S \rightarrow [0,1]$ é a função de transição de estados, onde $P(s'|s, a)$ denota a probabilidade de transitar ao estado s' ao executar a ação a no estado s ; $R: S \times A \rightarrow \mathbb{R}$ é a função de recompensa imediata esperada; e $\gamma \in [0,1]$ é o fator de desconto temporal.

A propriedade de Markov é central ao MDP: o estado futuro depende unicamente do estado e da ação atuais, sendo independente de toda a história anterior. Essa propriedade simplifica radicalmente a estrutura do problema, mas exige que o vetor de estado seja

suficientemente informativo, isto é, que capture toda a informação relevante do ambiente para a tomada de decisão (SUTTON; BARTO, 2018).

A Figura 4.1 sintetiza o loop de aprendizado por reforço (MDP) aplicado ao escalonamento proposto: o agente DDPG observa o estado da rede, executa uma ação contínua e recebe uma recompensa escalar do ambiente (HetNetEnv/Vienna 5G SLS).

Figura 4.1- Loop de aprendizado por reforço



Fonte: Elaborada pelo autor com base em Sutton e Barto (2018).

Em problemas parcialmente observáveis (POMDP, do inglês *Partially Observable Markov Decision Process*), o estado latente não é diretamente acessível, e a observação o_t pode ser apenas uma projeção ruidosa de s_t . A prática comum é incorporar ao vetor de observação variáveis de memória, como ações e estados anteriores, para aproximar a propriedade de Markov (ALWARAFY et al., 2022).

A cada instante t , o agente observa s_t , seleciona a ação a_t de acordo com uma política π , recebe $r_t = R(s_t, a_t)$ e transita para $s_{t+1} \sim P(\cdot | s_t, a_t)$. O objetivo é maximizar o retorno descontado, conforme a equação 4.1 (SUTTON; BARTO, 2018):

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}. \quad (4.1)$$

O fator de desconto γ controla a importância relativa de recompensas futuras: $\gamma \rightarrow 1$ corresponde a um agente com horizonte longo de planejamento, enquanto $\gamma \rightarrow 0$ torna o

agente míope. Em episódios finitos de comprimento T , o somatório é truncado e $G_t = \sum_{k=0}^{T-t-1} \gamma^k r_{t+k}$.

4.1.2 Funções de Valor e Equações de Bellman

As funções de valor quantificam a qualidade de estar em um estado ou de executar uma ação em um estado, sob uma política π . A função de valor de estado $V^\pi(s)$ e a função de valor de ação $Q^\pi(s, a)$ são definidas conforme as equações 4.2 e 4.3 (SUTTON; BARTO, 2018), respectivamente:

$$V^\pi(s) = E_\pi[G_t | s_t = s], \quad (4.2)$$

$$Q^\pi(s, a) = E_\pi[G_t | s_t = s, a_t = a]. \quad (4.3)$$

As equações de Bellman relacionam V^π e Q^π entre si, expressando o valor de um par (s, a) em função do valor imediato e do valor esperado do estado seguinte (SUTTON; BARTO, 2018):

$$Q^\pi(s, a) = R(s, a) + \gamma \sum_{s'} P(s' | s, a) V^\pi(s'). \quad (4.4)$$

A política ótima π^* maximiza Q^π para todo par (s, a) . A Q-função ótima $Q^*(s, a)$ satisfaz a equação de otimalidade de Bellman que é a condição necessária e suficiente para otimalidade (SUTTON; BARTO, 2018):

$$Q^*(s, a) = R(s, a) + \gamma \sum_{s'} P(s' | s, a) \max_{a'} Q^*(s', a'). \quad (4.5)$$

A política ótima é recuperada por $\pi^*(s) = \operatorname{argmax}_a Q^*(s, a)$. No *Q-learning*, a diferença entre o alvo Bellman e a estimativa corrente define o erro de diferença temporal (TD, do inglês *Temporal Difference*) δ_t , que dirige o processo de aprendizagem. A forma específica do alvo varia por algoritmo (SARSA usa a ação efetivamente tomada; métodos ator-crítico empregam o crítico), mas o princípio de correção incremental é compartilhado por todos os algoritmos de diferença temporal. Para o *Q-learning* especificamente:

$$\delta_t = r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t). \quad (4.6)$$

4.1.3 Q-Learning e Aprendizado Off-Policy

O *Q-Learning*, introduzido por Watkins (1989) e formalizado com prova de convergência por Watkins e Dayan (1992), é um algoritmo livre de modelo (*model-free*) que estima $Q^*(s, a)$ iterativamente a partir de experiências coletadas, sem necessidade de conhecer $P(s'|s, a)$. É classificado como *off-policy* porque utiliza a ação *greedy* $\arg\max_{a'} Q(s', a')$ no alvo de atualização, independentemente da ação que gerou a transição. Isso permite aprender a política ótima enquanto se segue uma política exploratória (ex.: ϵ -greedy). A regra de atualização é (WATKINS; DAYAN, 1992):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)], \quad (4.7)$$

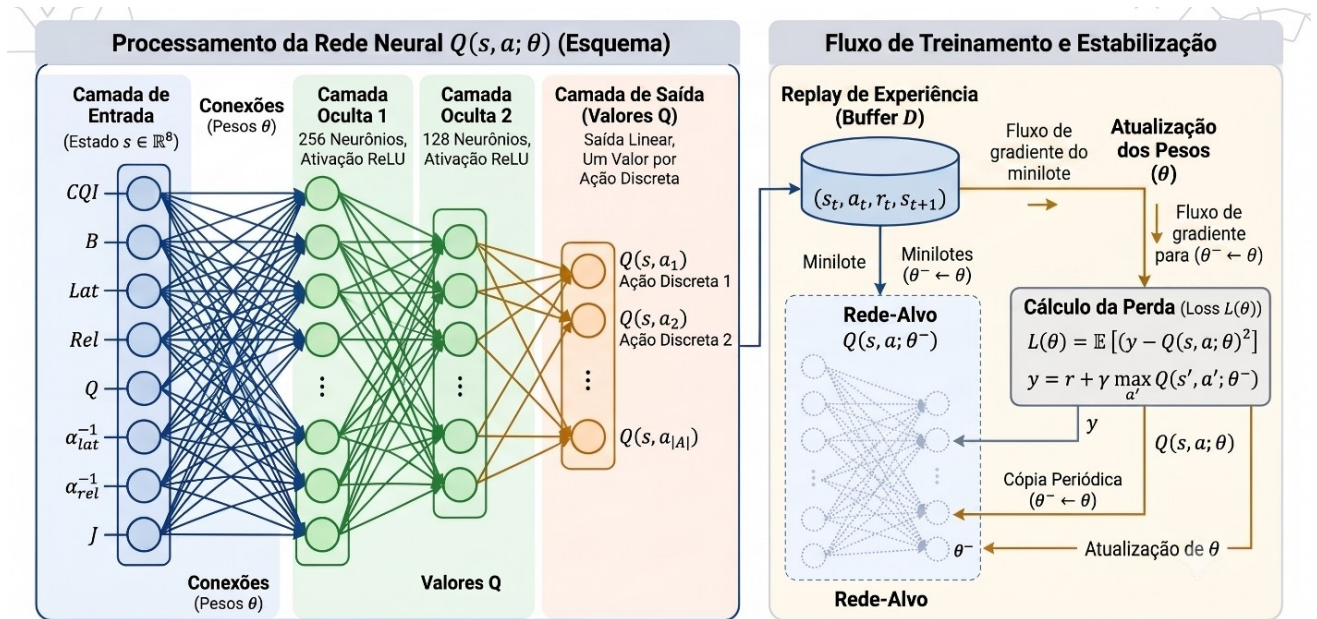
onde $\alpha \in (0, 1]$ é a taxa de aprendizado. Watkins e Dayan (1992) demonstraram que, sob as condições de Robbins-Monro ($\sum_t \alpha_t = \infty$ e $\sum_t \alpha_t^2 < \infty$) e desde que todos os pares (s, a) sejam visitados infinitamente, $Q(s, a) \rightarrow Q^*(s, a)$ com probabilidade 1. Em espaços de estado grandes ou contínuos, a representação tabular de $Q(s, a)$ torna-se inviável, motivando a substituição por aproximadores de função paramétricos.

4.2 Deep Q-Network (DQN)

A *Deep Q-Network* (DQN), proposta por Mnih et al. (2013) e consolidada em Mnih et al. (2015), substituiu a tabela Q por uma rede neural profunda $Q(s, a; \theta)$, com parâmetros θ ajustados por gradiente descendente estocástico. Isso viabilizou o aprendizado em espaços de estado de alta dimensão como imagens *raw* de jogos por exemplo o Atari, que seriam intratáveis com abordagens tabulares.

A Figura 4.2 ilustra a arquitetura da *Deep Q-Network* utilizada neste trabalho, com camada de entrada ($s \in \mathbb{R}^8$), duas camadas ocultas de 256 e 128 neurônios com ativação ReLU e camada de saída linear com um valor Q por ação discreta, como instância concreta do modelo de Mnih et al. (2015), adaptada ao espaço de estados e ações do problema de escalonamento proposto (detalhado no Capítulo 5).

Figura 4.2 - Arquitetura da Deep Q-Network (DQN)



Fonte: Elaborada pelo autor com base em Sutton e Barto (2018).

Entretanto, o treinamento direto de redes neurais com *Q-learning* era instável por dois motivos fundamentais: (i) a correlação temporal entre amostras consecutivas viola a suposição de i.i.d. necessária para a convergência do gradiente descendente estocástico; e (ii) a não estacionariedade do alvo TD, que se desloca junto com os parâmetros em otimização, criando um ciclo de retroalimentação que frequentemente diverge (MNIH et al., 2015). Duas inovações introduzidas pelo DQN resolveram esses problemas: o *experience replay*, que descorrelaciona as amostras de treinamento, e a rede-alvo, que estabiliza o alvo TD. Ambas são detalhadas nas Seções 4.2.1 e 4.2.2, respectivamente.

4.2.1 Experience Replay

O *experience replay* armazena as transições (s_t, a_t, r_t, s_{t+1}) produzidas durante a interação com o ambiente em um *buffer* circular D de capacidade N . A cada passo de treinamento, um *minibatch* de N_B transições é amostrado uniformemente de D e utilizado para computar o gradiente de $L(\theta)$ (MNIH et al., 2013).

Essa estratégia traz dois benefícios centrais. O primeiro é a quebra de correlações temporais: ao amostrar aleatoriamente de D , as transições no *minibatch* são praticamente descorrelacionadas, aproximando a suposição de i.i.d. O segundo é a reutilização de experiências: cada transição pode ser amostrada múltiplas vezes, melhorando a eficiência

amostral que é especialmente importante quando interações com o ambiente são custosas, como em simulações de nível de sistema (MNIH et al., 2013). Uma desvantagem do *replay buffer* uniforme é que todas as transições recebem igual probabilidade de seleção, independentemente de sua importância para o aprendizado. O *Prioritized Experience Replay* corrige isso amostrando com maior probabilidade transições com maior $TD - error|\delta_t|$, acelerando o aprendizado (LI, 2018).

4.2.2 Rede-Alvo

A rede-alvo $Q(s, a; \theta^-)$ é uma cópia periódica dos pesos da rede principal, atualizada a cada C passos de treinamento por $\theta^- \leftarrow \theta$. O alvo Bellman é calculado com θ^- fixo, evitando que o alvo mude a cada passo de gradiente (MNIH et al., 2015). A função de perda minimizada em cada iteração é dada pela equação 4.8:

$$L(\theta) = E_{(s,a,r,s') \sim D} \left[\left(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta) \right)^2 \right]. \quad (4.8)$$

Mantendo θ^- fixo durante a otimização de θ , o problema se assemelha ao aprendizado supervisionado com alvos estacionários, para o qual existem soluções robustas de gradiente descendente. A cada C passos, θ^- é sincronizado com θ , fazendo o alvo avançar em direção à estimativa mais recente. O valor de C é um hiperparâmetro: valores muito pequenos de C comprometem a estabilidade (alvo muda rápido demais); valores muito grandes retardam a propagação de informação de valor (MNIH et al., 2015).

4.2.3 Double DQN e Dueling DQN

O DQN original apresenta viés de superestimação: a operação $\max_{a'} Q(s', a'; \theta^-)$ usa as mesmas estimativas tanto para selecionar a melhor ação quanto para avaliá-la, introduzindo um viés positivo sistemático que se propaga recursivamente. O *Double DQN* (van Hasselt et al., 2016) corrige isso desacoplando seleção e avaliação: a ação é selecionada pela rede principal ($\arg \max_{a'} Q(s', a'; \theta)$) e avaliada pela rede-alvo, produzindo o alvo (VAN HASSELT et al., 2016):

$$y = r + \gamma Q(s', \arg \max_{a'} Q(s', a'; \theta); \theta^-). \quad (4.9)$$

O *Dueling* DQN (Wang et al., 2016) decompõe $Q(s, a; \theta)$ em dois estimadores paralelos: a função de valor de estado $V(s; \theta, \beta)$ e a função de vantagem $A(s, a; \theta, \alpha)$, combinados segundo (WANG et al., 2016):

$$Q(s, a; \theta, \alpha, \beta) = V(s; \theta, \beta) + [A(s, a; \theta, \alpha) - (1/|A|)\Sigma_{a'} A(s, a'; \theta, \alpha)]. \quad (4.10)$$

A centralização da vantagem pela sua média (em vez do máximo) garante que a decomposição seja identificável: V e A não podem compensar-se arbitrariamente. A vantagem prática é que V(s) é atualizado em cada passo independentemente da ação executada, acelerando o aprendizado do valor de estado em ambientes com muitas ações de valor similar (WANG et al., 2016).

Apesar dos avanços, DQN e suas variantes exigem que A seja discreto e finito, pois a operação $\operatorname{argmax}_a Q(s, a; \theta)$ deve ser computada explicitamente. Em problemas com ações contínuas como o ajuste de parâmetros de política de escalonamento, discretizar A em d dimensões com k valores por dimensão resulta em k^d ações totais, tornando inviáveis a exploração e o armazenamento da função Q. Esse é o obstáculo que motivou o desenvolvimento do DDPG (LILLICRAP et al., 2016).

4.3 Métodos de Gradiente de Política

Os métodos de gradiente de política otimizam diretamente os parâmetros θ de uma política parametrizada $\pi_\theta(a|s)$, maximizando o objetivo de desempenho $J(\theta) = E_{\tau \sim \pi_\theta}[G_0]$, onde $\tau = (s_0, a_0, r_1, \dots)$ é uma trajetória. A principal vantagem em relação aos métodos baseados em valor é a capacidade de tratar naturalmente espaços de ação contínuos, pois a política pode ser representada como uma distribuição parametrizada sobre ações reais (SUTTON; BARTO, 2018).

4.3.1 Teorema do Gradiente de Política

O teorema do gradiente de política (*Policy Gradient Theorem*) fornece uma expressão analítica para $\nabla_\theta J(\theta)$ sem necessidade de diferenciar a distribuição de estados $\rho^\pi(s)$ em relação a θ o que seria computacionalmente intratável (SUTTON; BARTO, 2018):

$$\nabla_{\theta} J(\theta) \propto E_{\pi_{\theta}} [\nabla_{\theta} \log \pi_{\theta}(a|s) \cdot Q^{\pi_{\theta}}(s, a)]. \quad (4.11)$$

A proporcionalidade decorre do fator de desconto embutido na distribuição de estados; em métodos episódicos, isto é, naqueles em que a interação com o ambiente se organiza em episódios de duração finita, com estado inicial e término bem definidos, a constante de proporcionalidade é absorvida pela taxa de aprendizado, de modo que a direção do gradiente é preservada. Na prática, $Q^{\pi_{\theta}}(s, a)$ é estimado pelo retorno amostrado G_t (método REINFORCE) ou por um crítico parametrizado $Q_w(s, a)$ (método ator-crítico). A arquitetura ator-crítico combina ambos: o ator π_{θ} define a política, e o crítico Q_w estima o valor de ação, reduzindo a variância do gradiente sem introduzir viés (SUTTON; BARTO, 2018). Para reduzir ainda mais a variância, utiliza-se a função de vantagem $A(s, a) = Q(s, a) - V(s)$ como sinal de treinamento do ator, que preserva o viés zero enquanto remove o componente de valor de estado que é independente da ação escolhida.

4.3.2 Gradiente de Política Determinístico

Silver et al. (2014) demonstraram que políticas determinísticas $\mu_{\theta}: S \rightarrow A$, que mapeiam cada estado a uma única ação real, também admitem um gradiente bem definido. O DPG para uma política μ_{θ} treinada *off-policy* é (SILVER et al., 2014):

$$\nabla_{\theta} J(\theta) = E_{s \sim \rho^{\beta}} [\nabla_{\theta} \mu_{\theta}(s) \cdot \nabla_a Q^{\mu_{\theta}}(s, a)|_{a=\mu_{\theta}(s)}], \quad (4.12)$$

onde $\rho^{\beta}(s)$ é a distribuição de estados induzida pela política de comportamento $\beta(a|s)$ que pode diferir de μ_{θ} para garantir exploração. A derivação aplica a regra da cadeia: $\nabla_{\theta} \mu_{\theta}(s)$ propaga o gradiente pelos parâmetros do ator, enquanto $\nabla_a Q(s, a)$ avalia como o valor Q muda em resposta a perturbações na ação.

Uma propriedade notável do DPG é sua eficiência computacional: enquanto o gradiente de política estocástico integra sobre o espaço de ações (requerendo N_A amostras por estado para estimativa precisa), o DPG integra apenas sobre o espaço de estados. Isso torna o DPG especialmente vantajoso em problemas com espaços de ação de alta dimensão, em que amostrar sobre A é custoso (SILVER et al., 2014). Silver et al. (2014) demonstraram ainda que o DPG é o caso limite do gradiente de política estocástico quando a variância da política $\pi_{\theta}(a|s) = N(\mu_{\theta}(s), \sigma^2 I)$ tende a zero ($\sigma \rightarrow 0$).

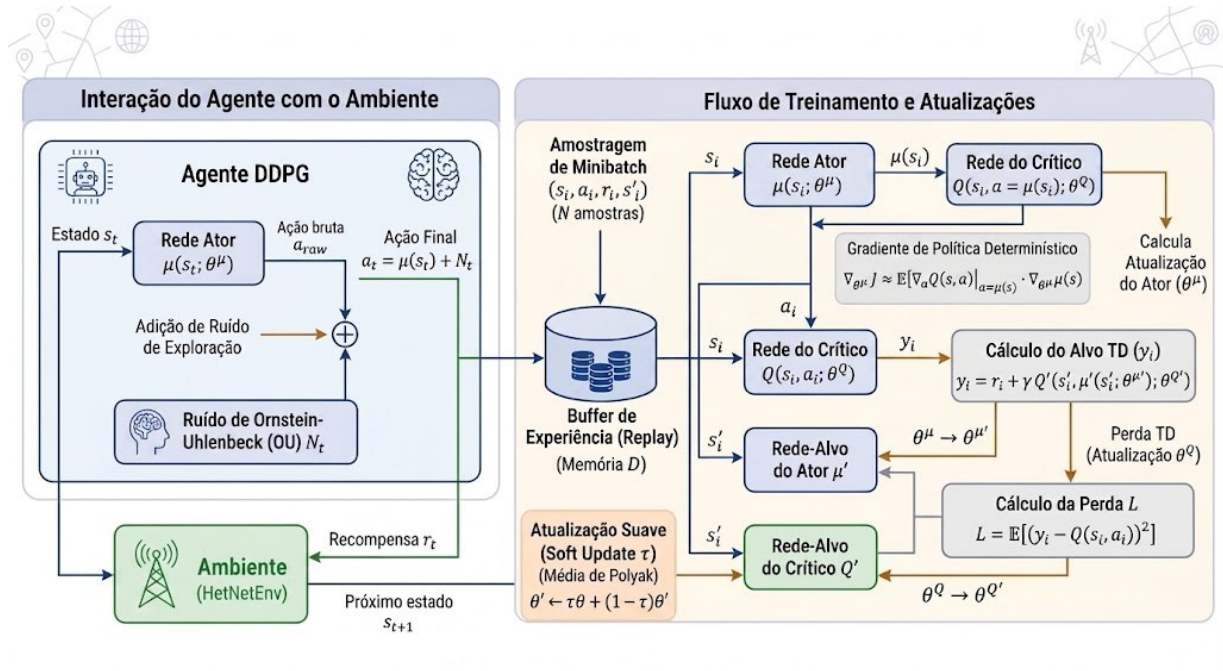
4.4 Deep Deterministic Policy Gradient (DDPG)

O *Deep Deterministic Policy Gradient* (DDPG), proposto por Lillicrap et al. (2016), integra o DPG de Silver et al. (2014) com as inovações do DQN (*experience replay* e redes-alvo) para aprender políticas determinísticas contínuas de forma estável e *off-policy*. O DDPG é classificado como um algoritmo ator-crítico livre de modelo (*model-free*), capaz de operar em espaços de ação contínuos de alta dimensão em que o DQN falha (LILLICRAP et al., 2016).

4.4.1 Estrutura do Algoritmo

O DDPG mantém quatro redes neurais: o ator $\mu(s; \theta^\mu)$ e o crítico $Q(s, a; \theta^Q)$, com seus respectivos pares de redes-alvo $\mu'(s; \theta^{\mu'})$ e $Q'(s, a; \theta^{Q'})$. O ator é uma rede que mapeia deterministicamente cada estado a uma ação contínua: $\mu: S \rightarrow A$. O crítico é uma rede que estima o valor Q do par (estado, ação), parametrizando tanto s quanto a como entradas (LILLICRAP et al., 2016). A Figura 4.3 apresenta a arquitetura ator-crítico resultante, com o fluxo de dados entre as quatro redes e o *buffer* de experiência.

Figura 4.3- Arquitetura Ator-Crítico Off-Policy DDPG



Fonte: Elaborada pelo autor com base em Lillicrap et al. (2016).

O fluxo de treinamento a cada passo segue quatro etapas. Primeiro, o agente executa a ação exploratória $a_t = \mu(s_t; \theta^\mu) + N_t$, onde N_t é o ruído de exploração, e armazena a transição (s_t, a_t, r_t, s_{t+1}) no *replay buffer* D. Segundo, um *minibatch* (s_i, a_i, r_i, s'_i) é amostrado de D. Terceiro, o crítico é atualizado minimizando a perda TD sobre o *minibatch*. Quarto, o ator é atualizado usando o DPG. As redes-alvo são atualizadas por *soft update* após cada passo (LILLICRAP et al., 2016).

4.4.2 Treinamento do Crítico

O crítico $Q(s, a; \theta^Q)$ é treinado minimizando o erro quadrático médio entre a estimativa $Q(s_i, a_i; \theta^Q)$ e o alvo Bellman y_i calculado com as redes-alvo (LILLICRAP et al., 2016):

$$L(\theta^Q) = (1/N) \sum_{i=1}^N [y_i - Q(s_i, a_i; \theta^Q)]^2, \quad (4.13)$$

$$y_i = r_i + \gamma Q'(s'_i, \mu'(s'_i; \theta^{\mu'}); \theta^{Q'}). \quad (4.14)$$

O alvo y_i emprega a política-alvo μ' e o crítico-alvo Q' , cujos parâmetros $\theta^{\mu'}$ e $\theta^{Q'}$ permanecem fixos durante a minimização de $L(\theta^Q)$, o que mantém o alvo estacionário em cada iteração de gradiente. A ação que compõe o alvo, $\mu'(s'_i; \theta^{\mu'})$, é obtida por avaliação direta da política-alvo, sem a maximização sobre o espaço de ações exigida pelo DQN, uma vez que μ' é um mapeamento determinístico de estado em ação (LILLICRAP et al., 2016).

4.4.3 Atualização do Ator

O ator $\mu(s; \theta^\mu)$ é atualizado aplicando o gradiente de política determinístico (Eq. 4.12) ao *minibatch* amostrado. Como $Q(s, a; \theta^Q)$ é diferenciável em relação a a , e $\mu(s; \theta^\mu)$ é diferenciável em relação a θ^μ , o gradiente é (LILLICRAP et al., 2016):

$$\nabla_{\theta^\mu} J \approx (1/N) \sum_i \nabla_a Q(s, a; \theta^Q)|_{s=s_i, a=\mu(s_i; \theta^\mu)} \cdot \nabla_{\theta^\mu} \mu(s; \theta^\mu)|_{s=s_i}. \quad (4.15)$$

O gradiente $\nabla_a Q(s, a; \theta^Q)$ indica a direção no espaço de ações que maximiza o valor Q . O produto com ∇_{θ^μ} , μ propaga esse direcionamento de volta para os parâmetros do ator pela regra da cadeia. Como θ^Q é mantido fixo durante a atualização de θ^μ , o problema equivale a

um passo de ascendente de gradiente do objetivo J ao longo dos parâmetros do ator (SILVER et al., 2014; LILLICRAP et al., 2016).

4.4.4 Soft Update das Redes-Alvo

O DDPG adota uma atualização suave (*soft update*, ou média de Polyak) para as redes-alvo, em vez da cópia periódica do DQN. A cada passo de treinamento, os parâmetros de ambas as redes-alvo são atualizados por (LILLICRAP et al., 2016):

$$\theta^{Q'} \leftarrow \tau \theta^Q + (1 - \tau) \theta^{Q'}, \theta^{\mu'} \leftarrow \tau \theta^{\mu} + (1 - \tau) \theta^{\mu'}, \quad (4.16)$$

com $\tau \ll 1$ (tipicamente na faixa de 10^{-3} a 10^{-2}). Valores pequenos de τ garantem que as redes-alvo evoluam lentamente em relação às redes principais, fornecendo alvos TD quase-estacionários ao longo de centenas de passos de treinamento. Comparado à cópia periódica do DQN, o *soft update* elimina os degraus nas estimativas de valor que ocorrem a cada sincronização, produzindo uma trajetória de aprendizagem mais suave e estável. Lillicrap et al. (2016) demonstraram empiricamente que ter redes-alvo independentes para o ator e o crítico e não apenas para o crítico, é necessário para obter alvos y_t estáveis e evitar divergência durante o treinamento.

4.4.5 Exploração por Ruído de Ornstein-Uhlenbeck

Como μ_{θ} é determinístico, a exploração do espaço de ações deve ser realizada externamente. O DDPG adiciona ruído estocástico à ação do ator durante o treinamento, formando a política de comportamento $\beta(s) = \mu(s; \theta^{\mu}) + N_t$ (LILLICRAP et al., 2016). A política de inferência (avaliação e implantação) continua sendo a política determinística pura μ .

O processo de Ornstein-Uhlenbeck (OU) é um processo estocástico de reversão à média, descrito pela equação diferencial estocástica (EDE) (LILLICRAP et al., 2016):

$$dN_t = \theta_{OU}(\mu_{OU} - N_t)dt + \sigma_{OU}dW_t, \quad (4.17)$$

onde $\theta_{OU} > 0$ é a taxa de reversão à média μ_{OU} (tipicamente $\mu_{OU} = 0$), $\sigma_{OU} > 0$ é a volatilidade e W_t é um processo de Wiener (movimento Browniano). A solução discreta desta EDE é $N_{t+1} = N_t + \theta_{OU}(\mu_{OU} - N_t)\Delta t + \sigma_{OU}\sqrt{\Delta t} \cdot \varepsilon_t$, onde $\varepsilon_t \sim N(0,1)$.

O processo OU é caracterizado por autocorrelação temporal positiva: o ruído em $t+1$ é próximo do ruído em t (suavizado pela reversão à média). Essa propriedade produz trajetórias de exploração suaves e fisicamente coerentes, ao contrário do ruído gaussiano i.i.d. que gera perturbações bruscas e descorrelacionadas. Em problemas de controle de parâmetros como o ajuste contínuo de pesos de escalonamento, trajetórias suaves de exploração são preferíveis porque permitem ao agente avaliar de forma consistente o efeito de uma perturbação antes de corrigi-la (LILLICRAP et al., 2016).

A amplitude do ruído de exploração é gradualmente reduzida ao longo do treinamento por meio de um coeficiente de decaimento $\sigma_t = \sigma_0 \cdot (1 - \lambda)^t$, onde λ é a taxa de decaimento. Isso garante exploração suficiente nas fases iniciais do treinamento, quando a política ainda é aleatória, e convergência para uma política mais determinística nas fases finais, quando o agente já aprendeu uma boa política base.

4.5 Funções de Recompensa Multiobjetivo

Em problemas de gerenciamento de recursos em redes de comunicação, o agente deve otimizar simultaneamente múltiplos objetivos potencialmente conflitantes: eficiência espectral, equidade, latência e confiabilidade. A abordagem clássica para tratar esses problemas no arcabouço do MDP é a escalarização por soma ponderada (*Scalarization*), que converte o vetor de objetivos $f(s, a) = [f_1(s, a), \dots, f_k(s, a)]$ num único escalar de recompensa (ALWARAFY et al., 2022):

$$r(s, a) = \sum_{i=1}^k w_i f_i(s, a), w_i \geq 0, \sum w_i = 1. \quad (4.18)$$

A escolha dos pesos w_i é um grau de liberdade do projetista que determina qual ponto da fronteira de Pareto será privilegiado. Um vetor de pesos w define um único ponto na fronteira; para explorar diferentes compromissos, múltiplos agentes podem ser treinados com vetores w distintos (ALWARAFY et al., 2022).

Um aspecto crítico do projeto de recompensa no DRL é a normalização dos objetivos individuais f_i . Grandezas em escalas muito distintas (ex.: *throughput* em Mbps vs. BLER em

$[0,1]$) podem causar dominância de gradiente: a rede aprende muito mais rápido sobre o objetivo de maior magnitude, negligenciando os demais. A normalização de cada f_i ao intervalo $[0,1]$ garante que os pesos w_i reflitam de fato as preferências do projetista (SUTTON; BARTO, 2018).

Outro aspecto importante é a regularização da política. Termos de penalidade adicionados à recompensa como $\|a - a_{ref}\|^2$ ou $\|a\|^2$ desincentivam ações extremas ou instáveis, promovendo políticas suaves e previsíveis. Essa regularização é especialmente relevante quando o espaço de ações parametriza um controlador heurístico: ações próximas aos limites do espaço $[0,1]^d$ podem levar o heurístico a comportamentos degenerados que afetam negativamente o desempenho da rede (ALWARAFY et al., 2022).

4.6 Ensemble de Agentes em DRL

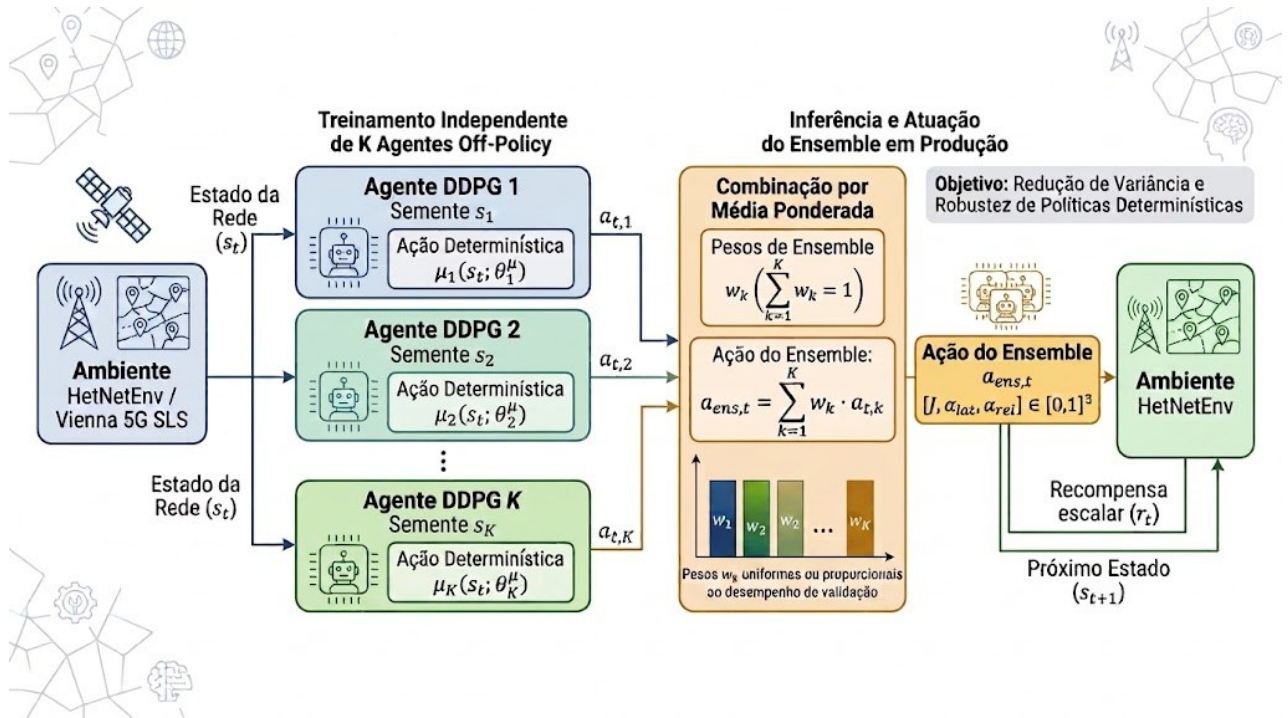
O treinamento de agentes DRL por gradiente descendente estocástico é inerentemente não determinístico: a sensibilidade à inicialização aleatória dos pesos θ_0 e às trajetórias de exploração geradas durante o treinamento resulta em políticas com desempenhos distintos entre execuções com sementes diferentes, mesmo que os hiperparâmetros sejam idênticos (LI, 2018). Isso levanta uma questão prática relevante: como selecionar e consolidar os melhores resultados de múltiplos treinamentos?

A estratégia de *ensemble* amplamente utilizada em aprendizado supervisionado para reduzir variância e aumentar robustez pode ser aplicada ao DRL combinando as políticas de K agentes independentes. Cada agente μ_k é treinado com uma semente distinta e produz uma política determinística $\mu_k(s; \theta_k^\mu)$. A ação do *ensemble* é a média ponderada das ações individuais (LI, 2018):

$$a_{ens}(s) = \sum_{k=1}^K w_k \cdot \mu_k(s; \theta_k^\mu), \sum w_k = 1. \quad (4.19)$$

Os pesos w_k podem ser uniformes ($w_k = 1/K$) ou proporcionais ao desempenho de cada agente em um conjunto de validação. Uma prática comum é avaliar cada agente em episódios de avaliação sem exploração e selecionar os K_{top} melhores para compor o *ensemble*, descartando os de baixo desempenho (LI, 2018). A Figura 4.4 ilustra a estratégia de *ensemble*, desde o treinamento independente dos K agentes até a agregação das ações individuais em uma única decisão.

Figura 4.4- Estratégia de Ensemble de Agentes DDPG



Fonte: Elaborada pelo autor.

A justificativa teórica do *ensemble* em DRL é análoga à do *bagging*: se os K agentes são suficientemente diversos (políticas não correlacionadas), a média ponderada converge para um estimador com variância menor do que qualquer agente individual. Na prática, a diversidade é garantida pelas diferentes sementes de inicialização e trajetórias de exploração. Uma limitação do *ensemble* por média de ações é que, em espaços de Q não-convexo, a média de boas ações pode produzir uma ação subótima, por exemplo, se dois agentes recomendam ações opostas em torno de um máximo local, a média cai num ponto subótimo. No presente trabalho, isso é mitigado pelo espaço de ação compacto $[0,1]^3$, geometricamente convexo, e pela seleção prévia das cinco políticas de melhor desempenho, estatisticamente equivalentes, sobre as quais se aplica média de pesos uniformes (Seção 5.8). O *ensemble* é particularmente benéfico em cenários com alta variabilidade espacial, como distribuições de Poisson de usuários, em que um agente individual pode ter desempenho instável em realizações desfavoráveis do cenário, enquanto a média de múltiplos agentes exibe comportamento consistente (LI, 2018).

4.7 Aplicações em Redes sem Fio

O DRL tem sido extensivamente aplicado ao gerenciamento de recursos de rádio (RRM) em redes 5G e HetNets. Alwarafy et al. (2022) realizam uma revisão sistemática das aplicações

de DRL em RRM, classificando os trabalhos por tipo de recurso (potência, espectro, acesso), tipo de rede e algoritmo DRL. Os autores identificam como limitação central dos métodos convencionais a incapacidade de lidar com a larga escala, a heterogeneidade e a não-estacionariedade dos ambientes reais, para as quais o DRL oferece soluções naturais.

Garbazza, Silva e Guardieiro (2025) aplicam DQN ao escalonamento *uplink* em redes LTE-A/5G heterogêneas, demonstrando que o agente aprende políticas superiores às heurísticas *Round Robin* e PF em termos de *throughput* e *fairness*. Entretanto, a discretização forçada do espaço de ação limita a granularidade das decisões. Wang et al. (2024) combinam D3QN com previsão de tráfego para alocação de blocos de recursos físicos (PRBs, do inglês *Physical Resource Blocks*) em sistemas 5G IoT com restrições de atraso por fatia, em uma arquitetura de dois níveis na qual o D3QN resolve a alocação inter-fatias e uma heurística gerencia a alocação intra-fatia, uma abordagem que compartilha a lógica de controle em camadas do presente trabalho.

Hu et al. (2023) propõem o DDPG-D3QN, uma arquitetura híbrida para sistemas de computação borda-nuvem com espaços de ação mistos (contínuo + discreto). O crítico adota uma estrutura *dual-path*, com fluxos separados para estado e ação antes da fusão, o que melhora a robustez da estimativa de Q quando as escalas das variáveis de entrada são heterogêneas. Naderalizadeh et al. (2021) desenvolvem um *framework* multi-agente descentralizado em que cada transmissor é um agente DRL independente que decide potência e usuário a servir, evidenciando a escalabilidade do DRL para cenários HetNets densos e a robustez a variações no número de agentes entre treinamento e teste.

A abordagem de controle indireto em que o agente DRL ajusta parâmetros estratégicos de um controlador heurístico base em vez de substituí-lo tem ganhado atenção crescente na literatura (ALWARAFY et al., 2022; WANG et al., 2024). Essa estratégia reduz a dimensão do espaço de ação, estabiliza o treinamento ao aproveitar o conhecimento de domínio embutido no heurístico e melhora a capacidade de generalização. É especialmente adequada para o problema de escalonamento com tráfego misto eMBB/uRLLC, cuja alta complexidade e variabilidade tornam o controle direto de recursos computacionalmente custoso e difícil de treinar de forma estável.

4.8 Posicionamento do Presente Trabalho

A Tabela 4.1 sintetiza as principais características dos trabalhos de DRL para RRM citados neste capítulo, posicionando o presente trabalho em relação ao estado da arte. O

DRLQoS Scheduler proposto se distingue pelo uso do simulador Vienna SLS v1.4 (permitindo reprodutibilidade em cenário padronizado 3GPP), pela ação de controle indireto de três parâmetros contínuos do QAS e pelo *ensemble* de agentes DDPG para redução de variância.

Tabela 4.1 - Comparação entre abordagens de DRL para escalonamento em redes sem fio

Trabalho	Algoritmo	Estado (dim.)	Ação	Recompensa	Simulador	Baselines
Garbazza et al. (2025)	DQN	<i>throughput, buffer</i> (discreto)	Alocação RB (disc.)	<i>Throughput + Fairness</i>	ns-3 (LTE-A)	RR, PF
Wang et al. (2024)	D3QN	PRB, atraso, fatia (multi-dim.)	Alocação inter-fatia (disc.)	Throughput + QoS	5G IIoT (não informado)	Heurístico fixo
Hu et al. (2023)	DDPG-D3QN	<i>Edge-cloud state</i> (misto)	Misto (cont.+disc.)	Latência + Energia	Borda/nuvem (não informado)	Baseline fixo
Naderializadeh et al. (2021)	Multi-agente DRL	Estado local (por agente)	Potência + usuário	<i>Throughput</i>	Multi-agente (customizado)	WMMSE, greedy
Este trabalho	DDPG (ensemble)	Vazão, equidade, prazo, BLER, ação anterior	Parâmetros de QoS e equidade (3D contínuo)	Vazão + equidade + prazo	Vienna SLS v1.4	RR, BestCQI, QAS, PF

Fonte: Elaborada pelo autor.

4.9 Considerações Finais

Este capítulo apresentou a base teórica do Aprendizado por Reforço Profundo, partindo dos fundamentos do MDP e do *Q-Learning* e avançando até as técnicas modernas de gradiente de política determinístico e o algoritmo DDPG. O MDP fornece o arcabouço matemático que formaliza o problema de escalonamento como uma sequência de decisões sob incerteza; as equações de Bellman definem as condições de otimalidade; e o DPG fornece a regra de atualização do ator que viabiliza políticas contínuas e determinísticas.

As técnicas de *experience replay*, rede-alvo e *soft update* introduzidas pelo DQN e adaptadas pelo DDPG são os pilares de estabilização que tornam o treinamento de redes neurais profundas viável em ambientes de AR. O ruído de Ornstein-Uhlenbeck garante exploração temporalmente coerente, adequada para problemas de controle contínuo.

A adoção do DDPG neste trabalho apoia-se em quatro razões. A primeira é o custo computacional: o algoritmo requer quatro redes neurais: ator, crítico e as respectivas redes-alvo,

orçamento compatível com o volume de interações exigido pelo simulador de nível de sistema. A segunda é a mitigação de variância por *ensemble*: treinar K agentes com sementes de inicialização distintas e agregar suas políticas reduz a variabilidade entre execuções, que é a principal limitação prática de métodos de gradiente de política determinístico. A terceira é a estabilidade conferida pela arquitetura de controle: o escalonador heurístico permanece como política de retaguarda, o que limita o impacto de decisões instáveis durante o treinamento. A quarta é a validação no domínio: estudos como Garbazza, Silva e Guardieiro (2025) e Hu et al. (2023) reportam convergência estável do DDPG em cenários de gerenciamento de recursos de rádio comparáveis.

A escalarização por soma ponderada fundamenta o projeto da função de recompensa multiobjetivo, com as considerações de normalização e regularização que garantem o aprendizado equilibrado de todos os objetivos. A estratégia de *ensemble* de agentes reduz a variabilidade inerente ao treinamento estocástico, produzindo políticas mais robustas para avaliação em cenários com alta variabilidade espacial. O Capítulo 5 detalha como todos esses elementos são integrados na implementação do DRLQoSScheduler.

5 SISTEMA PROPOSTO: DRLQOSSCHEDULER

Este capítulo apresenta o DRLQoSScheduler, escalonador de *downlink* multiobjetivo cujos pesos de QoS e cuja meta de equidade são ajustados em tempo de execução por um agente de Aprendizado por Reforço Profundo. A proposta responde diretamente à lacuna identificada no Capítulo 2 e às limitações dos escalonadores de referência discutidas na Seção 3.7: em redes heterogêneas com tráfego misto eMBB/uRLLC, qualquer ponderação fixa entre vazão, latência, confiabilidade e equidade é ótima apenas para uma faixa estreita de condições de carga, canal e distribuição espacial de usuários. Fora dessa faixa, o mesmo escalonador que maximiza a vazão agregada penaliza sistematicamente os usuários de borda, ou o que protege os prazos uRLLC desperdiça capacidade quando não há tráfego crítico pendente.

A contribuição central não é um novo algoritmo de alocação de RBs, mas um mecanismo de controle: manter a heurística de alocação determinística, com complexidade e comportamento previsíveis, e delegar ao agente apenas a escolha contínua dos três parâmetros que definem o ponto de operação no espaço de compromissos. Essa separação de responsabilidades é o que permite obter adaptabilidade sem abandonar as garantias de tempo de execução exigidas por um escalonador que decide a cada intervalo de transmissão.

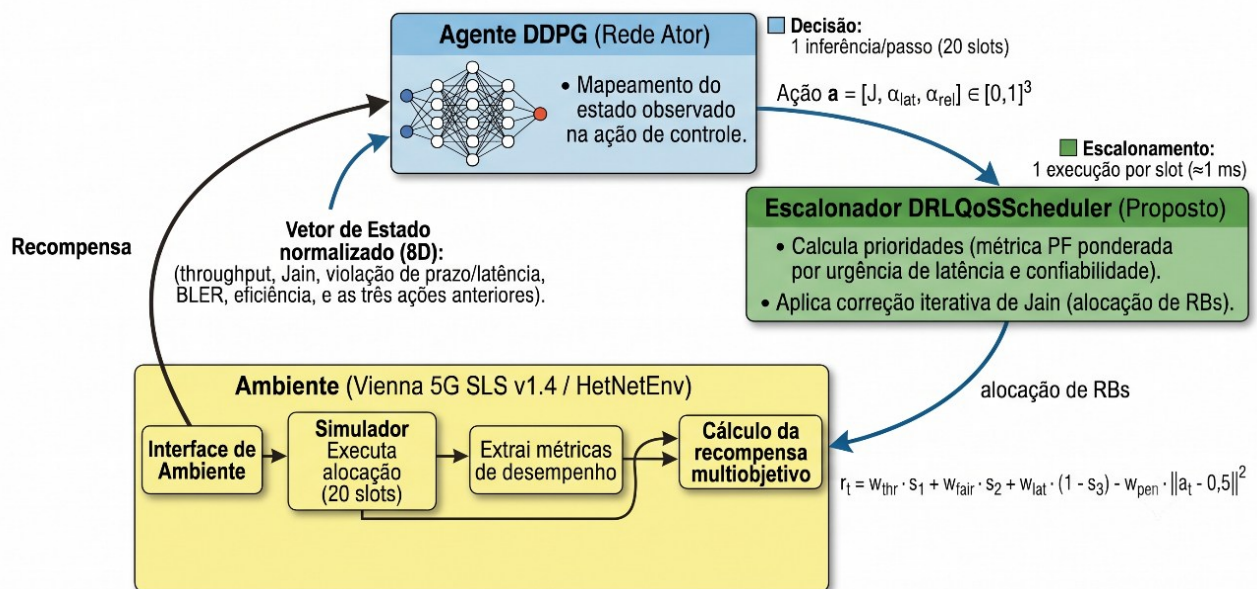
A implementação foi realizada sobre o *Vienna 5G System Level Simulator* versão 1.4 (MÜLLER et al., 2018), em MATLAB, integrando o simulador de nível de sistema ao *Reinforcement Learning Toolbox* por meio de uma interface de ambiente dedicada. As seções seguintes apresentam a arquitetura do sistema (Seção 5.1), o princípio de projeto do controle paramétrico indireto (Seção 5.2), a formulação do problema como Processo de Decisão de Markov (Seção 5.3), o escalonador propriamente dito (Seção 5.4), o cenário de simulação (Seção 5.5), a arquitetura das redes neurais e os hiperparâmetros (Seção 5.6), o protocolo de treinamento multiagente (Seção 5.7), a seleção do *ensemble* (Seção 5.8), o protocolo de avaliação (Seção 5.9), o custo de inferência (Seção 5.10) e as limitações metodológicas assumidas (Seção 5.11).

5.1 Arquitetura do Sistema

O sistema proposto organiza-se em três módulos com fronteiras bem definidas, articulados em um laço fechado de controle. O primeiro é o agente, uma política determinística contínua treinada por *Deep Deterministic Policy Gradient* (DDPG, Seção 4.4), que recebe um resumo normalizado do estado da rede e devolve um vetor de três parâmetros de controle. O segundo é a interface de ambiente, responsável por injetar esse vetor no simulador, executar um intervalo de simulação, extrair as métricas de desempenho resultantes e convertê-las em observação e recompensa no formato esperado pelo agente. O terceiro é o escalonador, que a cada intervalo de transmissão lê os parâmetros correntes e executa a alocação de RBs aos usuários ativos da célula.

A Figura 5.1 ilustra o laço de controle e o fluxo de informação entre os três módulos, evidenciando as duas escalas temporais em que o sistema opera: o escalonador decide a cada *slot* de 1 ms, enquanto o agente reavalia os parâmetros a cada bloco de 20 *slots*. Essa separação de escalas é deliberada e é discutida na Seção 5.3.4.

Figura 5.1 - Arquitetura do sistema proposto: laço fechado agente–ambiente–escalonador



Fonte: Elaborada pelo autor.

A Tabela 5.1 delimita as responsabilidades de cada módulo. O ponto essencial é que nenhuma decisão de alocação de recursos atravessa a fronteira do agente: o agente nunca escolhe qual usuário recebe qual RB. Ele apenas desloca o ponto de operação da heurística, que permanece inteiramente determinística. Consequentemente, uma política mal treinada degrada

o desempenho, mas não produz alocações inválidas, não viola a ortogonalidade da grade de recursos e não altera a complexidade de pior caso do escalonamento.

Tabela 5.1 - Responsabilidades dos módulos do sistema proposto.

Módulo	Implementação	Responsabilidade	Escala temporal
Agente	Rede ator DDPG (<i>ensemble</i> de 5 políticas)	Mapear o estado observado da rede no vetor de parâmetros de controle	Uma inferência por passo de decisão (20 slots)
Interface de ambiente	Classe derivada de ambiente MATLAB do <i>Reinforcement Learning Toolbox</i>	Injetar a ação no simulador, executar o intervalo, extrair métricas, compor observação e recompensa	Um passo por ação
Escalonador	Classe derivada do escalonador base do Vienna 5G SLS	Calcular prioridades, alocar blocos de recursos, aplicar correção de equidade e atualizar a memória de vazão	Uma execução por slot (1 ms)

Fonte: Elaborada pelo autor.

A injeção da ação no simulador merece uma nota de projeto. O Vienna 5G SLS instancia os escalonadores no início da simulação, a partir de um objeto de parâmetros; não há, no simulador original, um canal para alterar parâmetros de escalonamento durante a execução. A solução adotada explora o fato de que o contêiner de parâmetros de escalonamento é um objeto por referência: o escalonador guarda essa referência em seu construtor e, a cada *slot*, lê o campo que contém o vetor de ação corrente. A interface de ambiente escreve nesse campo antes de disparar cada intervalo de simulação. O acoplamento resultante é mínimo, isto é, um único campo adicional no contêiner de parâmetros, e não requer modificação do núcleo do simulador.

Um novo valor de enumeração foi acrescentado ao catálogo de tipos de escalonador do simulador e uma entrada correspondente foi adicionada à fábrica de escalonadores da classe base, de modo que o DRLQoSScheduler possa ser selecionado pelo mesmo mecanismo de configuração usado pelos escalonadores nativos. Essa escolha preserva a compatibilidade com o restante da infraestrutura do Vienna: os mesmos arquivos de cenário, o mesmo mecanismo de sobrescrita de parâmetros e as mesmas rotinas de coleta de resultados servem aos quatro escalonadores comparados no Capítulo 6.

5.2 Princípio de Projeto: Controle Paramétrico Indireto

Esta seção justifica as três decisões de projeto que definem a proposta: controlar parâmetros em vez de alocar recursos diretamente, usar exatamente três parâmetros de controle e adotar um espaço de ação contínuo.

5.2.1 Fundamentos para a Não Alocação Direta de RBs

A formulação mais direta de escalonamento como problema de aprendizado por reforço faria a ação ser a própria matriz de alocação, isto é, a atribuição de cada RB a um usuário. Essa formulação foi descartada por três razões.

- **Dimensionalidade e dependência da configuração:** Com N_{RB} RBs e N_{UE} usuários ativos, o espaço de ações discreto tem cardinalidade $N_{UE}^{N_{RB}}$. No cenário adotado nesta dissertação ($N_{RB} = 50$), esse número é intratável, mas o problema mais grave é estrutural: a dimensão da ação passaria a depender do número de usuários ativos, que varia de *slot* a *slot* e entre células. A política aprendida ficaria atrelada a uma configuração específica de rede e não transferiria para cenários com densidade diferente.
- **Orçamento de tempo por decisão:** O escalonador é executado uma vez por *slot*, por célula. Uma inferência de rede neural por *slot* e por célula, em um cenário com 21 setores macro e 16 nós de baixa potência, imporá um custo computacional incompatível com o intervalo de 1 ms, além de inviabilizar as simulações de 1 000 *slots* repetidas 30 vezes que compõem o protocolo de avaliação.
- **Perda de garantias:** Uma política que aloca diretamente pode, durante a exploração ou após convergência imperfeita, produzir alocações degeneradas, por exemplo, concentrar todos os blocos em um único usuário por longos períodos. Não há como impor, por construção, o comportamento proporcionalmente justo que o PF garante analiticamente.

O controle paramétrico indireto contorna as três dificuldades de uma só vez. A dimensão da ação é fixa e independente do número de usuários; a inferência ocorre uma vez a cada 20 *slots*, e não a cada *slot*; e a alocação continua sendo produzida por uma heurística cujo comportamento limite é conhecido. O preço pago é a restrição do espaço de políticas alcançáveis: o agente só pode atingir alocações que a heurística é capaz de produzir para algum valor dos parâmetros. Trata-se de uma restrição real, e a Seção 5.11 a discute como limitação. Em contrapartida, a proposta herda da heurística tanto a complexidade computacional quanto a

interpretabilidade das decisões, atributos que a literatura de escalonamento por DRL revisada no Capítulo 2 raramente oferece.

5.2.2 Parametrização dos Compromissos Multiobjetivo

Os três parâmetros correspondem exatamente aos três eixos de compromisso que o problema multiobjetivo de escalonamento com tráfego misto apresenta, conforme formulado na Seção 3.3.3, e cada um deles tem efeito monotônico e interpretável sobre a alocação.

- O primeiro parâmetro, denotado J , é a meta de equidade de Jain. Controla o eixo vazão agregada versus equidade entre usuários. Em $J \rightarrow 0$, o escalonador opera como uma heurística de vazão pura; em $J \rightarrow 1$, impõe distribuição tão uniforme quanto a alocação discreta de blocos permite.
- O segundo, α_{lat} , é o peso de urgência de latência. Controla o eixo vazão versus cumprimento de prazos. Em $\alpha_{lat} = 0$, os prazos de pacote são ignorados e o escalonador se comporta como um *Proportional Fair* clássico; em $\alpha_{lat} = 1$, reproduz a urgência máxima do QAS de referência.
- O terceiro, α_{rel} , é o peso de confiabilidade. Controla o eixo eficiência espectral versus proteção dos usuários com enlace degradado. Em $\alpha_{rel} = 0$, a taxa de erro de bloco histórica não influencia a prioridade; em $\alpha_{rel} = 1$, a penalização por taxa de erro é máxima.

Essa escolha tem uma propriedade metodológica importante: os escalonadores de referência são casos particulares do escalonador proposto. A configuração ($J \rightarrow 0, \alpha_{lat} = 0, \alpha_{rel} = 0$) recupera o comportamento de maximização de vazão; ($J = 0,7; \alpha_{lat} = 1; \alpha_{rel} = 1$) reproduz o QAS com pesos estáticos usado como linha de base. O espaço de políticas do agente contém, portanto, as linhas de base como pontos isolados, e qualquer ganho observado no Capítulo 6 é atribuível à *adaptação* dos parâmetros ao estado da rede, não a uma parametrização fixa mais favorável.

5.2.3 Espaço de Ação Contínuo e Escolha do Algoritmo

Os três parâmetros são grandezas naturalmente contínuas em $[0, 1]$, e a discretização introduziria um compromisso desfavorável: uma malha grosseira impediria ajustes finos próximos aos pontos de operação úteis, enquanto uma malha fina reintroduziria o problema de dimensionalidade com apenas 10 níveis por parâmetro, o espaço discreto já teria 1 000 ações. Além disso, evidências empíricas relatadas na Seção 3.7.5 indicam que o desempenho do escalonador é sensível a variações pequenas na meta de equidade, o que penaliza a discretização.

Por essa razão adotou-se o DDPG, algoritmo ator-crítico *off-policy* para espaços de ação contínuos, apresentado na Seção 4.4. O ator produz diretamente o vetor de três dimensões por meio de uma camada de ativação sigmoide, que garante por construção que a saída pertença a $[0, 1]^3$ o espaço de ação é respeitado pela arquitetura da rede, sem necessidade de projeção ou penalização.

5.3 Formulação como Processo de Decisão de Markov

O problema de ajuste dos parâmetros do escalonador é formalizado como o Processo de Decisão de Markov (S, A, P, R, γ) definido na Seção 4.1.1. Esta seção especifica cada elemento da tupla e justifica as escolhas de modelagem, em particular a normalização das observações, a estrutura da recompensa e o valor do fator de desconto.

5.3.1 Espaço de Estados

O espaço de estados é $S \subset [0, 1]^8$. A restrição ao hipercubo unitário é uma decisão de projeto, não uma propriedade do problema: métricas de rede têm escalas heterogêneas: vazão em Mbps, taxas em fração, índices adimensionais e a normalização evita que a rede neural precise aprender simultaneamente a escala e a política. Todas as componentes são saturadas em $[0, 1]$ antes de serem entregues ao agente, o que também impede que ruídos numéricos do simulador violem os limites declarados do espaço de observação.

As oito componentes dividem-se em duas famílias. As cinco primeiras são medições do estado da rede após o intervalo de simulação; as três últimas são a ação aplicada nesse intervalo. A Tabela 5.2 detalha a semântica, a origem e a normalização de cada uma.

Tabela 5.2 - Componentes do vetor de estado observado pelo agente.

Componente	Grandeza	Origem e normalização	Interpretação
s_1	Vazão média normalizada	Média da vazão de <i>downlink</i> sobre usuários e <i>slots</i> , dividida pela referência de 5,0 Mbps e saturada em 1	Eficiência agregada do escalonamento
s_2	Índice de equidade de Jain	Calculado sobre as vazões médias por usuário (Seção 3.4), já adimensional em $[1/N, 1]$	Uniformidade da distribuição de recursos
s_3	Taxa de violação de prazo	Fração das latências de transmissão registradas que excedem 60 slots	Grau de descumprimento dos requisitos do tráfego crítico
s_4	Taxa de erro de bloco média	Média da taxa de erro de bloco de <i>downlink</i> sobre usuários e slots	Confiabilidade agregada do enlace
s_5	Eficiência relativa ao limite de canal	Razão entre a vazão observada e a vazão de referência do escalonador de melhor canal, quando disponível na saída do simulador	Distância ao teto de eficiência espectral
s_6	Meta de equidade anterior	Primeira componente da ação aplicada no intervalo	Memória da decisão anterior
s_7	Peso de latência anterior	Segunda componente da ação aplicada	Memória da decisão anterior
s_8	Peso de confiabilidade anterior	Terceira componente da ação aplicada	Memória da decisão anterior

Fonte: Elaborada pelo autor.

A inclusão das três componentes de memória (s_6 a s_8) responde a uma característica fundamental do problema: o estado observado é um resumo agregado, e não o estado físico completo da rede, que incluiria os coeficientes instantâneos de canal, as ocupações de todos os *buffers* e as posições de todos os usuários. Trata-se, portanto, de um Processo de Decisão de Markov Parcialmente Observável, conforme a discussão da Seção 4.1.2. A concatenação da ação anterior ao vetor de observação implementa a técnica padrão de aumento de estado por histórico de janela unitária: o agente passa a distinguir, por exemplo, uma condição de equidade elevada obtida com meta baixa que indica canal homogêneo de uma equidade elevada obtida com meta alta, que indica atuação efetiva da correção.

A quinta componente exige uma qualificação de observabilidade. Ela é obtida do campo de vazão de referência do escalonador de melhor canal produzido pelo simulador, isto é, da vazão que seria alcançada alocando-se todos os blocos ao usuário de melhor indicador de qualidade de canal. Em uma rede real, essa referência não é diretamente mensurável: seria necessário estimá-la a partir dos relatórios de qualidade de canal disponíveis na estação-base, com sondagem periódica ou com o máximo indicador reportado no intervalo. A componente é, nesse sentido, a menos diretamente observável do vetor de estado, e a implementação adota um

valor neutro de 0,5 quando o campo não está disponível na saída da simulação. A Seção 5.1.1 retoma esse ponto.

5.3.2 Espaço de Ações

O espaço de ações é $A = [0, 1]^3$, com $a_t = [J, \alpha_{lat}, \alpha_{rel}]$. A Tabela 5.3 associa cada componente ao mecanismo do escalonador que ela governa e aos comportamentos-limite correspondentes, detalhados na Seção 5.4.

Tabela 5.3 - Componentes do vetor de ação e mecanismos controlados.

Componente	Mecanismo controlado	Limite inferior	Limite superior
J	Meta de equidade da correção iterativa de alocação (Seção 5.4.4)	Correção desativada: alocação puramente proporcional-justa ponderada por QoS	Redistribuição até o limite permitido pela granularidade do bloco de recursos
α_{lat}	Base efetiva do fator de urgência de latência (Seção 5.4.2)	Prazos ignorados: fator de urgência unitário para todos os usuários	Urgência máxima, equivalente ao QAS de referência
α_{rel}	Base efetiva do fator de urgência de confiabilidade (Seção 5.4.3)	Taxa de erro histórica não penaliza a prioridade	Penalização máxima por taxa de erro de bloco

Fonte: Elaborada pelo autor.

A ação é submetida a saturação em $[0, 1]$ tanto na saída do agente quanto na leitura pelo escalonador. Essa redundância é intencional: a saturação no agente protege contra o ruído de exploração, que pode empurrar a ação para fora do domínio durante o treinamento, e a saturação no escalonador garante que uma ação corrompida ou ausente não produza comportamento indefinido. Na primeira componente, o limite inferior efetivo é ligeiramente positivo, e não zero, para que a verificação de meta de equidade tenha significado numérico bem definido. Na ausência de ação válida, por exemplo, se a inferência falhar, o escalonador retorna à ação neutra $[0,7; 0,5; 0,5]$. Ela mantém a meta de equidade no valor padrão do escalonador de referência e situa os dois pesos de urgência no ponto médio de sua faixa, o que corresponde a metade da intensidade daquele escalonador: base de latência 1,05 em vez de 1,1 e base de confiabilidade 1,5 em vez de 2. Não se trata, portanto, da configuração estática do escalonador de referência, que é recuperada por $(J = 0,7; \alpha_{lat} = 1; \alpha_{rel} = 1)$, conforme a Seção 5.2.2, mas de um ponto de operação conservador, que não privilegia nenhum eixo do compromisso quando não há informação disponível.

5.3.3 Função de Recompensa

A recompensa é uma escalarização linear ponderada dos objetivos, acrescida de um termo de regularização da ação, conforme o princípio de conformação de recompensa discutido na Seção 4.5:

$$r_t = w_{thr} \cdot s_1 + w_{fair} \cdot s_2 + w_{lat} \cdot (1 - s_3) - \lambda \cdot \|a_t - 0,5\|^2, \quad (5.1)$$

com $w_{thr} = 0,30$, $w_{fair} = 0,35$, $w_{lat} = 0,35$ e $\lambda = 0,05$. O valor resultante é saturado no intervalo $[-0,1; 1,0]$, o que limita a variância dos alvos de aprendizado do crítico e evita que uma simulação anômala produza um retorno de magnitude desproporcional no *buffer* de experiência.

Três aspectos da Equação 5.1 requerem justificativa.

1. **Ponderação dos objetivos.** Os pesos atribuem prioridade conjunta a equidade e cumprimento de prazos (0,35 cada) sobre vazão agregada (0,30). A escolha reflete o argumento central da dissertação: em tráfego misto, a vazão agregada é a métrica mais facilmente maximizável e a menos informativa sobre a qualidade percebida, pois um escalonador de melhor canal a maximiza enquanto viola sistematicamente prazos e penaliza usuários de borda. Os pesos somam 1,0 e a soma dos três termos de objetivo pertence a $[0, 1]$, de modo que a escala da recompensa é comparável entre configurações. A sensibilidade da política a essa ponderação é uma questão legítima, cujo aprofundamento fica registrado entre os trabalhos futuros.
2. **Ausência de termo explícito de confiabilidade.** A taxa de erro de bloco é observada pelo agente (s_4), mas não figura na recompensa, ainda que α_{rel} seja uma componente da ação. A razão é evitar dupla contagem: o efeito de α_{rel} sobre a taxa de erro é indireto e opera pela realocação de recursos em favor de usuários com enlace degradado, o que reduz a taxa de erro média mas também eleva a vazão do percentil inferior e, portanto, o índice de equidade já presente na recompensa. Incluir a taxa de erro como quarto termo criaria um sinal

parcialmente redundante com o termo de equidade, ampliando a variância do gradiente sem acrescentar informação de controle. O agente aprende a ajustar α_{rel} pelos efeitos que esse ajuste produz nos termos existentes. Registre-se, contudo, que a eficácia desse mecanismo está condicionada a uma limitação da integração atual, documentada na Seção 5.4.3.

3. **Termo de regularização.** A penalização quadrática do desvio da ação em relação ao centro do domínio cumpre duas funções. Numericamente, desencoraja a saturação prematura da camada sigmoide do ator, cujo gradiente se anula nos extremos e produz estagnação do aprendizado. Do ponto de vista de controle, penaliza ações extremas quando elas não trazem ganho mensurável nos objetivos, induzindo preferência por pontos de operação intermediários mais robustos a variações de cenário. O coeficiente $\lambda = 0,05$ mantém a penalização máxima em 0,0375, isto é, abaixo de 4% da faixa da recompensa: a regularização molda a política sem dominar o sinal de objetivo.

5.3.4 Dinâmica de Transição e Fator de Desconto

A função de transição $P(s_{t+1} | s_t, a_t)$ não tem forma fechada: ela é definida implicitamente pela composição da injeção da ação nos parâmetros do escalonador com a execução de 20 *slots* de simulação de nível de sistema. Cada passo de decisão corresponde, portanto, a 20 ms de operação da rede, durante os quais o escalonador executa 20 vezes com os mesmos parâmetros. Um episódio compreende 20 passos, totalizando 400 *slots* simulados, e termina por contagem de passos, sem condição de parada antecipada.

A escolha de 20 *slots* por passo equilibra dois requisitos opostos. Um passo muito curto tornaria as métricas observadas dominadas por realizações instantâneas de desvanecimento, e o agente aprenderia a reagir a ruído; um passo muito longo reduziria a granularidade do controle e o número de transições coletadas por unidade de tempo de simulação, que é o recurso escasso neste trabalho. Vinte *slots* correspondem a algumas dezenas de coerências temporais de canal no cenário adotado, o suficiente para que as médias sejam estáveis.

O fator de desconto adotado é $\gamma = 0,95$. Esse valor merece justificativa explícita, porque a arquitetura poderia sugerir que cada passo de decisão constitui um problema isolado, sem acoplamento temporal com os passos vizinhos. Sob essa leitura, $\gamma = 0$ seria adequado, uma vez que cada passo executa uma nova simulação. Não é o que ocorre aqui, por dois mecanismos concretos de persistência de estado *dentro* do episódio:

- A memória exponencial de vazão mantida pelo escalonador para o cálculo da métrica proporcional-justa não é reinicializada entre passos. Ela acumula o histórico de alocações ao longo de todos os *slots* do episódio, de modo que a ação aplicada em um passo altera as prioridades relativas observadas nos passos subsequentes. Com fator de suavização de 0,05, a constante de tempo dessa memória é de aproximadamente 20 *slots*, exatamente a duração de um passo, o que garante acoplamento entre passos consecutivos.
- A ação anterior compõe as três últimas dimensões do estado. A decisão tomada em t influencia a observação em $t + 1$ por construção, criando dependência temporal explícita na trajetória.

Um valor de desconto próximo de 1 é, portanto, coerente com a estrutura do problema. Com $\gamma = 0,95$ e horizonte de 20 passos, o horizonte efetivo é de aproximadamente 20 passos, o que corresponde à duração do episódio: o agente é levado a considerar as consequências de suas ações ao longo de todo o episódio, sem que o desconto seja tão próximo de 1 que torne a estimativa de valor instável em um horizonte finito. Entre episódios o ambiente é reinicializado, o que restaura a ação neutra e o contador de passos e devolve a observação padrão [0,5; 0,7; 0,1; 0,1; 0,6; 0,7; 0,5; 0,5], escolhida por representar uma condição de rede tipicamente intermediária. O Algoritmo 5.1 sintetiza o passo de interação entre agente e simulador, incluindo a injeção da ação, a extração de métricas e a composição da observação e da recompensa.

Algoritmo 5.1 — Passo de interação entre o agente e o simulador

```

Entrada: ação a = [J, alpha_lat, alpha_rel]
Saída: observação s, recompensa r, indicador de término d

1 a <- satura(a, 0, 1)
2 incrementa contador de passos do episódio
3 escreve a no campo de ação do contêiner de parâmetros do escalonador
4 resultado <- executa simulação do cenário por 20 slots
5 (thr, jain, latViol, bler, effRel) <- extrai métricas de resultado
6 s <- satura([thr, jain, latViol, bler, effRel, a], 0, 1)
7 r <- w_thr*thr + w_fair*jain + w_lat*(1 - latViol)
8 r <- r - lambda * ||a - 0,5||^2
9 r <- satura(r, -0,1, 1,0)
10 d <- (contador de passos >= 20)
11 retorna (s, r, d)

```

Fonte: Elaborada pelo autor.

A extração de métricas na linha 5 opera sobre as estruturas de resultado nativas do simulador e adota valores neutros quando um campo não está disponível, o que confere robustez ao laço de treinamento: uma simulação que falhe não interrompe o treinamento de horas, apenas contribui com uma transição pouco informativa. Esse cuidado é relevante dado que o protocolo de treinamento descrito na Seção 5.7 executa aproximadamente 10 000 simulações por agente.

5.4 O Escalonador DRLQoSScheduler

O DRLQoSScheduler estende a classe base de escalonadores do Vienna 5G SLS e reimplementa a rotina de escalonamento de *downlink*. Ele parte da estrutura do escalonador sensível a QoS de referência (SHIYAHIN, 2022), descrito na Seção 3.7.5, e introduz três modificações: os pesos de QoS deixam de ser fixos e passam a ser escalados pela ação do agente; a métrica de prioridade incorpora explicitamente uma memória proporcional-justa de vazão; e uma etapa de correção iterativa de equidade é aplicada após a alocação inicial.

O Algoritmo 5.2 apresenta a estrutura da rotina executada a cada *slot*. As subseções seguintes detalham e justificam cada etapa.

Algoritmo 5.2 — Escalonamento de downlink por slot no DRLQoSScheduler

```

Entrada: slot corrente t; usuários associados à célula
Saída: atribuição de cada bloco de recursos a um usuário

1 reinicia a grade de recursos das antenas da estação-base
2 se não há usuários associados: delega ao tratamento comum e retorna
3 (J, alpha_lat, alpha_rel) <- satura a ação corrente lida dos parâmetros
4 U <- usuários ativos (com dados pendentes em buffer)
5 se U vazio: delega ao tratamento comum e retorna
6 T[u,r] <- vazão estimada do usuário u no bloco r, a partir do CQI reportado
7 u_rel[u] <- fator de urgência de confiabilidade (alpha_rel)
8 u_lat[u] <- fator de urgência de latência (alpha_lat, prazo, idade do pacote)
9 urg[u] <- u_rel[u] * u_lat[u]
10 para cada bloco r:
11 score[u,r] <- urg[u] * T[u,r] / thrMedio[u]
12 aloc[r] <- argmax_u score[u,r]
13 aloc <- corrigeEquidade(aloc, T, J)
14 escreve aloc na grade de recursos das antenas
15 thrMedio[u] <- (1 - beta) * thrMedio[u] + beta * vazaoObtida[u]
16 delega ao tratamento comum de sinalização e HARQ

```

Fonte: Elaborada pelo autor.

5.4.1 Métrica de Prioridade Proporcional-Justa Ponderada por QoS

A decisão de alocação repousa sobre a métrica calculada na linha 11 do Algoritmo 5.2:

$$score(u, r) = u_{rel}(u) \cdot u_{lat}(u) \cdot T(u, r) / T(\bar{u}), \quad (5.2)$$

em que $T(u, r)$ é a vazão instantânea estimada para o usuário u no RB r , obtida do indicador de qualidade de canal reportado e da tabela de eficiência espectral do simulador, e $T(\bar{u})$ é a média exponencial da vazão obtida pelo usuário nos *slots* anteriores.

A razão $T(u, r) / T(\bar{u})$ é exatamente a métrica do *Proportional Fair* apresentada na Seção 3.7.3: ela privilegia o usuário cuja condição instantânea de canal é boa *em relação ao que ele tem recebido*, o que produz a propriedade de equidade proporcional. Escolheu-se essa métrica como base em vez da vazão instantânea pura usada pelo escalonador de melhor canal porque ela já embute uma noção de justiça temporal e, sobretudo, porque introduz memória entre *slots*. Essa memória é o mecanismo que torna o problema um processo de decisão sequencial e não um problema de decisão isolada, conforme argumentado na Seção 5.3.4.

A contribuição do escalonador proposto é a multiplicação dessa métrica pelo produto dos dois fatores de urgência, cujo cálculo é controlado pela ação do agente. O produto, e não a soma, foi adotado por duas razões: preserva a homogeneidade dimensional da métrica proporcional-justa, dado que os fatores de urgência são adimensionais e pertencem a $(0, 1]$; e faz com que a urgência atue como um *modulador multiplicativo* da prioridade, de modo que um usuário com canal muito ruim não é promovido acima de todos os demais apenas por ter um prazo próximo. Uma combinação aditiva permitiria que o termo de urgência dominasse a métrica em regimes de canal degradado, produzindo alocações de baixa eficiência espectral.

A alocação inicial é obtida por decisão independente por RB: cada bloco é atribuído ao usuário de maior pontuação naquele bloco. Isoladamente, essa regra é ótima, porque a métrica é separável entre blocos; ela deixa de sê-lo quando se impõem as restrições que acoplam os blocos entre si, em particular a de equidade e a de atendimento aos *buffers*. A formulação exata é, então, um problema de atribuição binária com restrições acopladas, que o escalonador de referência resolve por relaxação contínua seguida de arredondamento, com auxílio de um solucionador externo de otimização convexa. A opção adotada aqui é deliberada: dispensa-se essa dependência, e o mecanismo de correção de equidade descrito na Seção 5.4.4 recupera parte da qualidade perdida.

5.4.2 Fator de Urgência de Latência

O fator de urgência de latência implementa escalonamento sensível a prazo segundo o mecanismo discutido na Seção 3.6.1, com a base da exponencial escalada pela ação do agente:

$$u_{lat}(u) = \sigma_{eff}^{-\max(d_c(u)-d(u),0)}, \text{ com } \sigma_{eff} = 1 + \alpha_{lat} \cdot (\sigma_{base} - 1), \quad (5.3)$$

em que $d_c(u)$ é o orçamento de atraso de pacote do fluxo do usuário, expresso em *slots*, e $d(u)$ é o atraso *head-of-line* do usuário, isto é, o tempo decorrido desde a geração do pacote mais antigo ainda retido em *buffer*. Ambos seguem as definições do arcabouço de QoS do 5G apresentado no Capítulo 3: o orçamento de atraso corresponde ao *Packet Delay Budget* associado ao 5QI do fluxo, tratado na Seção 3.4.1, e o atraso *head-of-line* é o definido na Seção 3.4.2. Já $\sigma_{base} = 1,1$ é o valor de base fixado na configuração do cenário. A quantidade $\max(d_c(u) - d(u), 0)$ é a margem residual do orçamento de atraso, truncada em zero: o expoente corresponde à parte positiva dessa margem, o que satura o fator na unidade, um usuário cujo orçamento de atraso já foi excedido recebe o mesmo valor daquele cuja margem se anulou no instante corrente, sem diferenciação pela magnitude da violação.

A leitura correta da Equação 5.3 é a de uma despriorização proporcional à diferença, em *slots*, entre o orçamento de atraso do fluxo e o tempo que o pacote mais antigo já aguarda em *buffer*, e não a de uma promoção proporcional ao atraso acumulado. Quando esse pacote é recém-gerado, a diferença é máxima e o fator assume seu valor mínimo, reduzindo a prioridade do usuário; à medida que o tempo de espera do pacote aumenta, a diferença diminui e o fator cresce monotonicamente até a unidade, alcançada quando o orçamento de atraso se esgota ou já foi excedido. Usuários cujo fluxo não declara orçamento de atraso, caso do tráfego de *buffer* cheio adotado para o perfil eMBB, mantêm fator unitário. O tráfego crítico, como o uRLLC, com a qualificação da Seção 5.5.3, é portanto temporariamente cedido em favor do tráfego elástico enquanto seu orçamento de atraso ainda não se esgotou, e recuperado com prioridade máxima quando esse esgotamento se aproxima. Esse comportamento é exatamente o desejável em um regime de tráfego misto: evita reservar capacidade para pacotes que ainda podem esperar.

O papel de α_{lat} é controlar a *intensidade* desse mecanismo. Em $\alpha_{lat} = 0$, tem-se $\sigma_{eff} = 1$ e o fator é unitário para todos os usuários, independentemente da folga: o escalonador degenera em um *Proportional Fair* puro e ignora completamente os prazos. Em $\alpha_{lat} = 1$, tem-se $\sigma_{eff} = \sigma_{base}$ e o mecanismo opera com a intensidade do escalonador de referência. Valores intermediários interpolam continuamente entre esses regimes. A relevância de dispor dessa interpolação como variável de controle é direta: quando não há tráfego crítico com prazo próximo, a urgência apenas distorce a métrica proporcional-justa e reduz a eficiência

espectral, e o agente pode desativá-la; quando o tráfego crítico se acumula, o agente pode elevá-la sem que isso exija reconfiguração manual do escalonador.

5.4.3 Fator de Urgência de Confiabilidade

O fator de confiabilidade segue a mesma estrutura, com a base escalada por α_{rel} e o expoente proporcional à taxa de erro de bloco histórica do usuário:

$$\begin{cases} u_{rel}(u) = \alpha_{eff}^{-BLER(u)} & \text{para } BLER(u) \leq 0,1; \\ u_{rel}(u) = 1 & \text{caso contrário.} \end{cases} \quad (5.4)$$

$\text{com } \alpha_{eff} = 1 + \alpha_{rel} \cdot (\beta_{base} - 1),$

em que $\beta_{base} = 2,0$ é a base de configuração do cenário. A formulação reproduz o mecanismo do escalonador de referência: usuários cuja taxa de erro histórica excede o limiar de 0,1 são excluídos da penalização e recebem fator unitário, ao mesmo tempo que são sinalizados para operar com um índice de qualidade de canal mais conservador; os demais recebem um desconto multiplicativo exponencial na taxa de erro. O efeito combinado é um leve ganho de prioridade relativa para os usuários de enlace degradado. Em $\alpha_{rel} = 0$, tem-se $\alpha_{eff} = 1$ e o fator é unitário para todos, anulando o mecanismo.

Duas limitações da integração desse mecanismo devem ser registradas explicitamente, pois condicionam a interpretação dos resultados do Capítulo 6:

- O histórico de taxa de erro não é realimentado: O vetor de taxa de erro por usuário é inicializado com o valor 0,1 no construtor do escalonador e não é atualizado a partir do modelo de desempenho de enlace ao longo da simulação. Esse comportamento é herdado da implementação de referência do QAS disponível na distribuição do simulador utilizada. Como consequência, a condição da Equação 5.4 é satisfeita por todos os usuários em todos os *slots* e o fator de confiabilidade assume o mesmo valor $\alpha_{eff}^{-0,1}$ para todos. Sendo um fator comum a todos os usuários, ele não altera o resultado da comparação por RB da linha 12 do Algoritmo 5.2: na configuração atual, α_{rel} não influencia a alocação.
- A sinalização de índice de qualidade conservador é inerte: O ajuste correspondente na classe base do simulador está condicionado a uma verificação

de tipo que reconhece apenas o QAS de referência. Como o escalonador proposto deriva diretamente da classe base, e não daquele, o ajuste não é aplicado.

Longe de comprometer a proposta, essas duas observações demonstram o rigor com que o alcance dos resultados foi delimitado. A Equação 5.4 é matematicamente consistente e recupera o comportamento de referência nos limites; o que as observações estabelecem é que os ganhos do Capítulo 6 se atribuem, de forma inequívoca, às dimensões de equidade e de latência da ação; nenhum resultado é creditado a um mecanismo que ainda não atua. A dimensão de confiabilidade está integralmente implementada e validada nos limites, e sua ativação plena depende apenas de uma correção circunscrita: realimentar a taxa de erro por usuário a cada *slot*, a partir das estatísticas do modelo de desempenho de enlace, e estender a verificação de tipo na classe base. Trata-se de uma extensão direta, registrada entre os trabalhos futuros.

5.4.4 Correção Iterativa de Equidade de Jain

A alocação gulosa por RB maximiza a métrica da Equação 5.2, mas não oferece controle direto sobre a *distribuição* resultante de vazão. Em redes heterogêneas, a potência de transmissão difere em mais de duas ordens de grandeza entre os tipos de célula macro, pico e femto, o que produz canais fortemente diversificados entre usuários. Nessas condições, a métrica proporcional-justa pode ainda assim concentrar recursos em poucos usuários. A etapa de correção existe para dar ao agente uma alavanca direta sobre essa distribuição.

O índice de equidade de Jain de uma alocação, definido na Seção 3.4, é calculado sobre as vazões esperadas por usuário decorrentes da alocação corrente. A correção compara esse índice com a meta J fornecida pelo agente e, enquanto a meta não é atingida, transfere RBs do usuário mais favorecido para o menos favorecido. O Algoritmo 5.3 detalha o procedimento.

Algoritmo 5.3 — Correção iterativa de equidade de Jain

```

Entrada: alocação  $\text{aloc}$ , vazões estimadas  $T$ , meta de equidade  $J$ 
Saída: alocação corrigida
1 para  $\text{iter} = 1$  até  $N_{RB}$ :
2  $\text{thr}[u] \leftarrow$  soma de  $T[u, r]$  sobre os blocos  $r$  atribuídos a  $u$ 
3  $F \leftarrow (\text{soma thr})^2 / (N_{UE} * \text{soma thr}^2)$  // índice de Jain
4 se  $F \geq J$ : interrompe // meta atingida
5  $\text{pobre} \leftarrow \text{argmin}_u \text{thr}[u]$ ;  $\text{rico} \leftarrow \text{argmax}_u \text{thr}[u]$ 
6 se  $\text{pobre} = \text{rico}$ : interrompe
7  $R \leftarrow$  blocos atribuídos a  $\text{rico}$ 
8  $\text{ganho}[r] \leftarrow T[\text{pobre}, r] - T[\text{rico}, r]$  para  $r$  em  $R$ 
9  $r^* \leftarrow \text{argmax}_r \text{ganho}[r]$ 
10 se  $\text{ganho}[r^*] \leq 0$ : interrompe // transferência não compensa
11  $\text{aloc}[r^*] \leftarrow \text{pobre}$  // transfere o bloco
12 retorna  $\text{aloc}$ 

```

Fonte: Elaborada pelo autor.

Três propriedades do procedimento merecem destaque:

1. **Terminação garantida:** O laço é limitado a N_{RB} iterações, o que impõe no máximo uma transferência por RB e assegura tempo de execução determinístico independentemente da configuração de canal. Além do limite estrutural, três condições de parada antecipada operam na prática: meta atingida (linha 4), degeneração para um único usuário (linha 6) e ausência de transferência benéfica (linha 10).
2. **Critério de transferência consciente da eficiência:** A escolha do bloco a transferir, nas linhas 8 e 9, não é arbitrária: seleciona-se o bloco em que o ganho do usuário desfavorecido supera *ao máximo* a perda do usuário favorecido. A condição da linha 10 impede transferências de ganho líquido não positivo. A correção, portanto, só se afasta da alocação de máxima métrica quando há ganho agregado de vazão. A equidade resulta da exploração da diversidade de canal entre blocos, e não do sacrifício cego de eficiência. Essa é a diferença essencial em relação a uma imposição de cota por usuário.
3. **Ineficácia identificada sob correlação forte de canal:** A condição da linha 10 falha quando o usuário favorecido apresenta vazão estimada superior à do desfavorecido em *todos* os blocos que detém, o que ocorre quando os ganhos de canal dos dois usuários são fortemente correlacionados em frequência. Nesse regime, a correção termina imediatamente sem realizar transferências e a meta J perde efeito. Trata-

se de uma limitação estrutural do critério adotado, cujo efeito se reflete nos resultados de equidade discutidos no Capítulo 6.

5.4.5 Atualização da Memória de Vazão

Após a aplicação da alocação à grade de recursos, a memória de vazão que alimenta a métrica proporcional-justa é atualizada por média móvel exponencialmente ponderada:

$$T(\bar{u}) \leftarrow (1 - \beta) \cdot T(\bar{u}) + \beta \cdot T_{obtida}(u), \quad (5.5)$$

com fator de suavização $\beta = 0,05$ e $T_{obtida}(u)$ igual à soma das vazões estimadas nos blocos efetivamente atribuídos ao usuário no *slot*. O vetor é inicializado com valor unitário para todos os usuários, o que evita divisão por zero na primeira avaliação da Equação 5.2 e faz com que, nos primeiros *slots*, a métrica se aproxime da vazão instantânea pura, comportamento aceitável durante o transitório inicial.

A escolha de $\beta = 0,05$ define uma constante de tempo de aproximadamente 20 *slots*. O valor não é arbitrário: ele coincide deliberadamente com a duração de um passo de decisão do agente, de modo que o efeito de uma ação sobre a distribuição de vazão se manifeste integralmente dentro do passo em que é aplicada, e permaneça parcialmente visível no passo seguinte. Um fator maior tornaria a métrica reativa a realizações instantâneas de desvanecimento, aumentando a variância das observações; um fator menor prolongaria a memória além do horizonte do episódio e tornaria a atribuição de crédito temporal mais difícil para o agente.

5.4.6 Complexidade Computacional

A complexidade por *slot* decompõe-se conforme a Tabela 5.4. A estimação de vazão e o cálculo da métrica são $O(N_{RB} \cdot N_{UE})$; a alocação gulosa é $O(N_{RB} \cdot N_{UE})$; a correção de equidade custa $O(N_{RB} \cdot N_{UE})$ por iteração e admite no máximo N_{RB} iterações, o que resulta em $O(N_{RB}^2 \cdot N_{UE})$ no pior caso.

Tabela 5.4 - Complexidade computacional por slot.

Etapa	Complexidade	Observação
Estimação de vazão por bloco	$O(N_{RB} \cdot N_{UE})$	Consulta à tabela de eficiência espectral
Cálculo da métrica e alocação gulosa	$O(N_{RB} \cdot N_{UE})$	Uma comparação máxima por bloco
Correção de equidade	$O(N_{RB^2} \cdot N_{UE})$ no pior caso	Termina antecipadamente na maioria dos slots
Atualização da memória de vazão	$O(N_{RB} + N_{UE})$	Contagem e média móvel
Inferência da política	Amortizada em 20 slots	Não ocorre a cada slot; ver Seção 5.10

Fonte: Elaborada pelo autor.

O termo dominante é o pior caso da correção de equidade. Com a configuração de grade adotada ($N_{RB} = 50$) e cerca de 40 usuários por célula, esse pior caso corresponde a aproximadamente 10^5 operações elementares por *slot*, ordem de grandeza compatível com o intervalo de 1 ms em hardware de estação-base, e não proibitiva em simulação. Na prática as condições de parada antecipada reduzem substancialmente esse custo.

O ponto relevante para a defesa da proposta é comparativo: a complexidade é da mesma ordem que a do QAS de referência, ao qual se acrescenta apenas o termo da correção de equidade. Nenhum solucionador de otimização convexa é invocado, ao contrário do que ocorre em parte das formulações revisadas no Capítulo 2, e a inferência da rede neural é amortizada ao longo de 20 *slots* em vez de executada por *slot*. É essa combinação que sustenta a afirmação de que o controle paramétrico indireto entrega adaptabilidade a custo computacional comparável ao das heurísticas clássicas.

5.5 Cenário de Simulação

Esta seção especifica o cenário implementado no Vienna 5G SLS. A configuração segue as recomendações de 3GPP TR 38.901 para ambiente urbano macro no que se refere a distância entre estações, alturas de antena e potências de transmissão, e adota as simplificações documentadas na Seção 5.11.

5.5.1 Parâmetros de Radiofrequência e Grade de Recursos

Os parâmetros de camada física correspondem à configuração de portadora do simulador, resumida na Tabela 5.5. A numerologia é a de espaçamento de subportadora de 15

kHz, correspondente a $\mu = 0$ em 5G NR, com *slot* de 1 ms, uma escolha coerente com a faixa de 2 GHz adotada e com o modelo de canal de macrocélula urbana.

Tabela 5.5 - Parâmetros de radiofrequência e da grade de recursos.

Parâmetro	Valor	Observação
Frequência central	2 GHz	Faixa sub-6 GHz
Largura de banda	5 MHz	Fração utilizável de 90%
Espaçamento de subportadora	15 kHz	Numerologia $\mu = 0$
Duração do <i>slot</i>	1 ms	Intervalo de decisão do escalonador
Blocos de recursos em frequência	25	Derivado da banda utilizável
Blocos de recursos em tempo por slot	2	Blocos de 0,5 ms
Total de blocos de recursos por slot	50	Recursos disponíveis por decisão
Tabela de índice de qualidade de canal	64-QAM, 15 níveis	Adaptação de enlace conforme Seção 3.2.2
Atraso de realimentação	3 slots	Latência do relatório de canal
Antenas por setor	1 transmissora	Configuração de fluxo único

Fonte: Elaborada pelo autor com base na configuração do Vienna 5G SLS v1.4.

5.5.2 Topologia Heterogênea Multicamada

A região de interesse é um quadrado de 1000 m de cada lado, conforme mostrado na Figura 5.2 dimensionada para acomodar integralmente a malha hexagonal de macrocélulas: com um anel de células e distância entre estações de 500 m, a rede macro ocupa aproximadamente essa área. Uma região menor cortaria células de borda e distorceria as estatísticas de interferência.

A rede compreende três tipos de célula com potências separadas por duas ordens de grandeza, o que produz o gradiente de cobertura característico das redes heterogêneas discutido na Seção 3.5.1 e é justamente a fonte da heterogeneidade de canal que torna o problema de escalonamento difícil. A Tabela 5.6 detalha os três tipos.

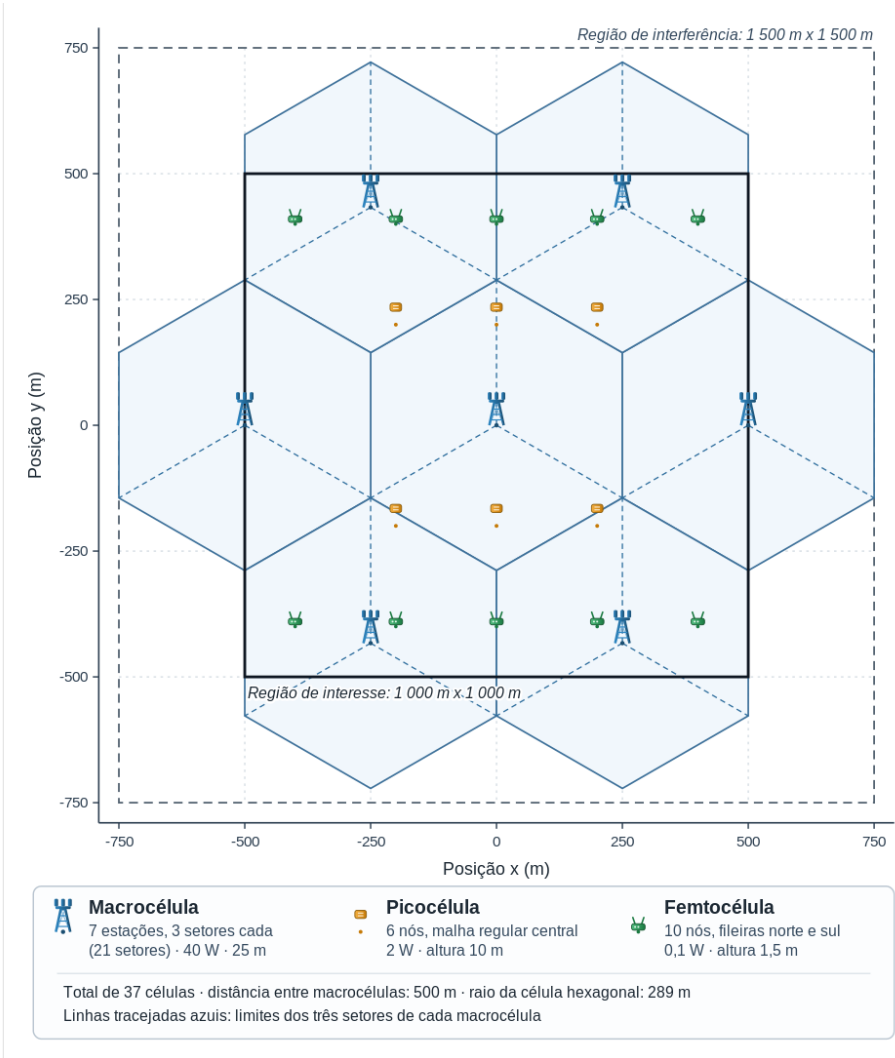
Uma configuração preliminar do cenário atribuía 12 W aos nós pico e 1 W aos femto, valores acima das faixas usuais para esses equipamentos. Como consequência, a diferença de potência entre os tipos de célula ficava comprimida, o que atenuava justamente o fenômeno que a proposta se propõe a explorar. Os valores adotados na Tabela 5.6 correspondem às faixas típicas de nós pico e femto.

Tabela 5.6 - Camadas da rede heterogênea

Camada	Disposição	Nós	Potência	Altura	Referência
Macro	Malha hexagonal, 1 anel, 3 setores, 500 m entre estações	7 estações (21 setores)	40 W (≈ 46 dBm)	25 m	3GPP TR 38.901, macro urbana
Pico	Posições predefinidas em malha regular na região central	6 nós	2 W (≈ 33 dBm)	10 m	Instalação externa típica
Femto	Posições predefinidas em duas fileiras nas bordas norte e sul	10 nós	0,1 W (≈ 20 dBm)	1,5 m	Célula de baixa potência

Fonte: Elaborada pelo autor.

Figura 5.2 - Disposição espacial das estações-base na topologia heterogênea multicamada



Fonte: Elaborada pelo autor.

A topologia descrita nesta seção é comum aos quatro cenários de avaliação. O cenário de estresse difere dos demais exclusivamente pela densidade de usuários, e preserva a distância

entre macrocélulas, a composição das camadas de baixa potência, o total de 37 células, a largura de banda e o prazo do perfil sensível à latência. A restrição é deliberada e tem propósito metodológico: com a topologia e o prazo congelados, toda diferença de desempenho observada em relação ao cenário de referência é atribuível à carga, e a nenhum outro fator. A configuração completa dos quatro cenários apresentados no Capítulo 6.

5.5.3 Perfis de Tráfego

Dois perfis de usuário coexistem, ambos distribuídos por processo pontual de Poisson bidimensional sobre a região de interesse, conforme a Tabela 5.7. A razão de densidades de 3:1 em favor do perfil eMBB reflete um cenário urbano denso, em que a maioria dos usuários demanda vazão elevada e uma minoria exige baixa latência.

Tabela 5.7 - Perfis de tráfego

Atributo	Perfil eMBB	Perfil sensível à latência
Densidade espacial	30 usuários por km ²	10 usuários por km ²
Usuários esperados na região	≈ 30	≈ 10
Modelo de tráfego	Buffer cheio	Tráfego interativo de jogos
Restrição de prazo	Não declarada (infinita)	60 slots
Tamanho de pacote	Ilimitado por construção	Distribuição de valores extremos, média ≈ 120 bytes
Intervalo entre pacotes	Não aplicável	Distribuição de valores extremos, média ≈ 55 ms

Fonte: Elaborada pelo autor.

Uma qualificação de nomenclatura é necessária e é mantida de forma consistente ao longo desta dissertação. O prazo de 60 *slots* corresponde a aproximadamente 60 ms na numerologia adotada, valor adequado a tráfego interativo de jogos e controle em tempo real, mas distante do requisito canônico de uRLLC definido pelo IMT-2020, da ordem de 1 ms. O simulador na versão utilizada não disponibiliza modelo de tráfego com prazo sub-milissegundo, e o termo uRLLC é empregado aqui para designar o perfil de tráfego sensível à latência disponível na plataforma, não o serviço uRLLC estrito. O prazo de 60 *slots* funciona como *proxy* de latência: preserva a estrutura do problema dois perfis com requisitos conflitantes, um deles com prazo rígido sem reproduzir a magnitude numérica do requisito normativo. Essa é uma limitação de validade externa, retomada na Seção 5.11.

5.5.4 Cenários de Treinamento e de Avaliação

Foram construídas duas variantes do cenário, idênticas em geometria, potências e perfis de usuário, mas distintas em duração e em refinamento de modelagem. A separação decorre de uma restrição prática incontornável: o treinamento exige da ordem de 10000 simulações por agente, e uma simulação de 1000 *slots* com região de interferência ativa consome vários minutos. A Tabela 5.8 explicita as diferenças.

Tabela 5.8 - Diferenças entre os cenários de treinamento e de avaliação.

Aspecto	Cenário de treinamento	Cenário de avaliação
Duração	1 bloco de 20 slots	10 blocos de 100 slots (1 000 slots)
Região de interferência	Desativada	Ativada, fator de extensão 1,5
Perda de percurso — macro	Macro célula urbana 5G	Macro célula urbana 5G com distinção de visada
Perda de percurso — pico	Macro célula urbana 5G	Espaço livre
Perda de percurso — femto	Macro célula urbana 5G	Cânion urbano, com e sem visada
Tempo aproximado por simulação	≈ 15 s	≈ 5 min

Fonte: Elaborada pelo autor.

A validade da transferência de política entre os dois cenários repousa em um argumento explícito. O agente não observa perda de percurso, potência ou geometria; observa exclusivamente o vetor de oito métricas normalizadas de desempenho. Um refinamento da modelagem de propagação altera os valores dessas métricas, mas não a estrutura do mapeamento estado-ação que a política representa: uma condição de equidade baixa continua exigindo elevação da meta de equidade, qualquer que seja a causa física da desigualdade. A política aprendida é, nesse sentido, uma regra de controle sobre métricas agregadas, e não um modelo do canal. O argumento não constitui garantia formal de transferibilidade; a verificação empírica ocorre no Capítulo 6, em que o desempenho no cenário de avaliação é comparado com as linhas de base sob o mesmo protocolo.

A escolha de 20 *slots* para o cenário de treinamento também tem justificativa estatística, além da computacional: com 20 *slots* por passo e 20 passos por episódio, cada episódio expõe o agente a 400 *slots* de operação, e os 500 episódios de treinamento acumulam 200 000 *slots* simulados por agente. A variabilidade de canal e de tráfego amostrada é, portanto, substancial, mesmo que cada passo individual observe uma janela curta.

5.6 Arquitetura das Redes Neurais e Hiperparâmetros

O agente foi implementado com o algoritmo DDPG apresentado na Seção 4.4, utilizando as construções de rede e de agente do *Reinforcement Learning Toolbox* do MATLAB. Esta seção documenta as escolhas de arquitetura e de hiperparâmetros e as razões que as motivaram.

5.6.1 Rede do Ator

O ator é uma rede totalmente conectada com três camadas ocultas de 256, 128 e 64 unidades, ativação linear retificada nas camadas ocultas e ativação sigmoide na camada de saída de três unidades. A estrutura afunilada é uma escolha convencional para políticas de baixa dimensionalidade: as primeiras camadas expandem a representação de oito métricas em um espaço de características mais rico, e as seguintes comprimem progressivamente até a decisão.

A ativação sigmoide na saída é a decisão de projeto mais relevante desta subseção. Ela garante por construção que a ação pertença a $[0, 1]^3$, isto é, que os limites do espaço de ação sejam respeitados pela arquitetura, sem necessidade de projeção posterior, penalização por violação ou saturação expedientes que introduzem descontinuidades no gradiente ou distorcem o sinal de aprendizado. O ônus dessa escolha é a possibilidade de saturação: nas extremidades do domínio o gradiente da sigmoide se anula e o aprendizado estagna. O termo de regularização da recompensa, introduzido na Equação 5.1, é a contramedida adotada para esse risco, o que ilustra a interdependência entre a escolha de arquitetura e a formulação da recompensa.

5.6.2 Rede do Crítico

O crítico estima a função de valor de ação e adota arquitetura de dois caminhos: um ramo processa o estado por duas camadas de 256 e 128 unidades, outro processa a ação por uma camada de 128 unidades, e os dois são combinados por soma elemento a elemento antes de duas camadas finais de 64 e 1 unidade.

A separação dos caminhos antes da fusão é a prática recomendada para DDPG e tem razão específica no problema em questão: estado e ação têm dimensionalidade muito desigual (oito contra três) e naturezas semânticas distintas. A concatenação direta na entrada faria as três componentes da ação competirem por capacidade representacional com as oito do estado desde a primeira camada, o que tende a reduzir a sensibilidade do valor estimado à ação. Como o

gradiente da política é obtido justamente pela derivada do valor em relação à ação (Seção 4.4.3), essa sensibilidade é o sinal de aprendizado do ator: perdê-la compromete o algoritmo. O ramo dedicado à ação assegura que ela seja processada com capacidade própria antes da fusão. A Tabela 5.9 consolida a topologia das duas redes.

Tabela 5.9 - Arquitetura das redes do ator e do crítico.

Rede	Entrada	Camadas ocultas	Saída	Ativações
Ator	Estado (8)	256 → 128 → 64	Ação (3)	Retificada nas ocultas, sigmoide na saída
Crítico — ramo de estado	Estado (8)	256 → 128	—	Retificada
Crítico — ramo de ação	Ação (3)	128	—	Linear
Crítico — ramo comum	Soma dos dois ramos	64	Valor (1)	Retificada após a soma e na camada oculta

Fonte: Elaborada pelo autor.

5.6.3 Hiperparâmetros de Treinamento

A Tabela 5.10 lista os hiperparâmetros adotados. Os valores seguem, em sua maioria, as recomendações originais do DDPG, com três desvios deliberados que requerem justificativa.

Tabela 5.10 - Hiperparâmetros do agente e do treinamento

Hiperparâmetro	Valor	Justificativa
Taxa de aprendizado do ator	10^{-4}	Uma ordem de grandeza abaixo do crítico
Taxa de aprendizado do crítico	10^{-3}	Crítico deve convergir mais rápido que o ator
Limite de norma do gradiente — ator	1	Contenção estrita: passos grandes na política desestabilizam a coleta
Limite de norma do gradiente — crítico	5	Folga maior: alvos de valor têm escala variável no início
Fator de desconto	0,95	Ver justificativa na Seção 5.3.4
Fator de suavização das redes-alvo	0,005	Atualização suave conforme Seção 4.4.4
Capacidade do buffer de experiência	20 000 transições	Cobre 50 episódios completos
Tamanho do minilote	128	Compromisso entre variância e custo por passo
Desvio-padrão inicial do ruído de exploração	0,2	20% da faixa de cada dimensão da ação
Taxa de decaimento do ruído	10^{-4}	Ajustada para decair ao longo do orçamento de episódios

Hiperparâmetro	Valor	Justificativa
Desvio-padrão mínimo do ruído	0,02	Mantém exploração residual até o fim do treinamento
Episódios por agente	500	Orçamento uniforme (ver Seção 5.7)
Passos por episódio	20	Equivale a 400 slots simulados
Janela de média da recompensa	50 episódios	Suavização para monitoramento de convergência

Fonte: Elaborada pelo autor.

- 1) Assimetria das taxas de aprendizado. O crítico aprende dez vezes mais rápido que o ator. Essa é a prática consolidada em métodos ator-crítico e tem razão direta: o ator ascende no gradiente de uma estimativa produzida pelo crítico. Se o ator se move rápido sobre uma estimativa ainda ruim, ele persegue um alvo espúrio e a coleta de dados degrada, realimentando o erro. Manter o crítico à frente é uma condição prática de estabilidade
- 2) Limites de gradiente diferenciados. A contenção mais estrita no ator reflete a mesma assimetria: uma atualização brusca da política altera imediatamente a distribuição das transições coletadas, com efeito realimentado, enquanto uma atualização brusca do crítico afeta apenas a qualidade momentânea da estimativa. Na configuração preliminar utilizava-se um limite único; a diferenciação reduziu a incidência de degradação abrupta da recompensa durante o treinamento.
- 3) Decaimento do ruído de exploração. O ruído de exploração é essencial em DDPG, cuja política é determinística e não explora por si. A taxa de decaimento foi calibrada para que o desvio-padrão decaia de 0,2 para próximo do piso de 0,02 ao longo dos 10 000 passos que compõem o orçamento de 500 episódios. Em uma configuração preliminar a taxa era duas ordens de grandeza menor, o que mantinha exploração intensa até o final do treinamento e impedia a política de consolidar um ponto de operação; a correção desse valor foi uma das intervenções de maior efeito sobre a convergência.

5.7 Protocolo de Treinamento Multiagente

Políticas obtidas por DDPG são conhecidamente sensíveis à semente de inicialização: execuções independentes do mesmo algoritmo, sobre o mesmo ambiente e com os mesmos hiperparâmetros, podem convergir para políticas de desempenho apreciavelmente distintas. Reportar o resultado de uma única execução é, por isso, metodologicamente frágil. O protocolo adotado treina dez agentes independentes com sementes documentadas, o que permite tanto

caracterizar a variabilidade do processo de aprendizado quanto compor um *ensemble*, conforme a motivação apresentada na Seção 4.6.

Cada agente recebe uma semente distinta, aplicada ao gerador de números pseudoaleatórios antes da construção das redes, de modo que a inicialização dos pesos, a amostragem de minilotes e a geração do ruído de exploração sejam reproduzíveis. O *buffer* de experiência é reinicializado entre agentes, assegurando independência estatística: sem essa reinicialização, transições geradas por uma política anterior contaminariam o treinamento seguinte e as dez execuções não seriam amostras independentes.

O Algoritmo 5.4 resume o protocolo completo, do treinamento à composição do *ensemble*.

Algoritmo 5.4 — Treinamento multiagente, ranqueamento e composição do ensemble

```

Entrada: conjunto de sementes {s_1,..., s_10}; cenário de treinamento
Saída: ensemble com K = 5 políticas e pesos uniformes

// Fase 1 – treinamento independente
1 para cada agente i = 1 até 10:
2   fixa gerador pseudoaleatório com a semente s_i
3   constrói redes do ator e do crítico e o agente DDPG
4   reinicia o buffer de experiência
5   treina por 500 episódios de 20 passos no cenário de treinamento
6   salva checkpoint a cada 100 episódios
7   salva a política final e a curva de recompensa por episódio

// Fase 2 – ranqueamento sem exploração
8 para cada agente i = 1 até 10:
9   executa 10 episódios com a política determinística (sem ruído)
10  score[i] <- média das recompensas acumuladas por episódio
11  desvio[i] <- desvio-padrão entre os 10 episódios

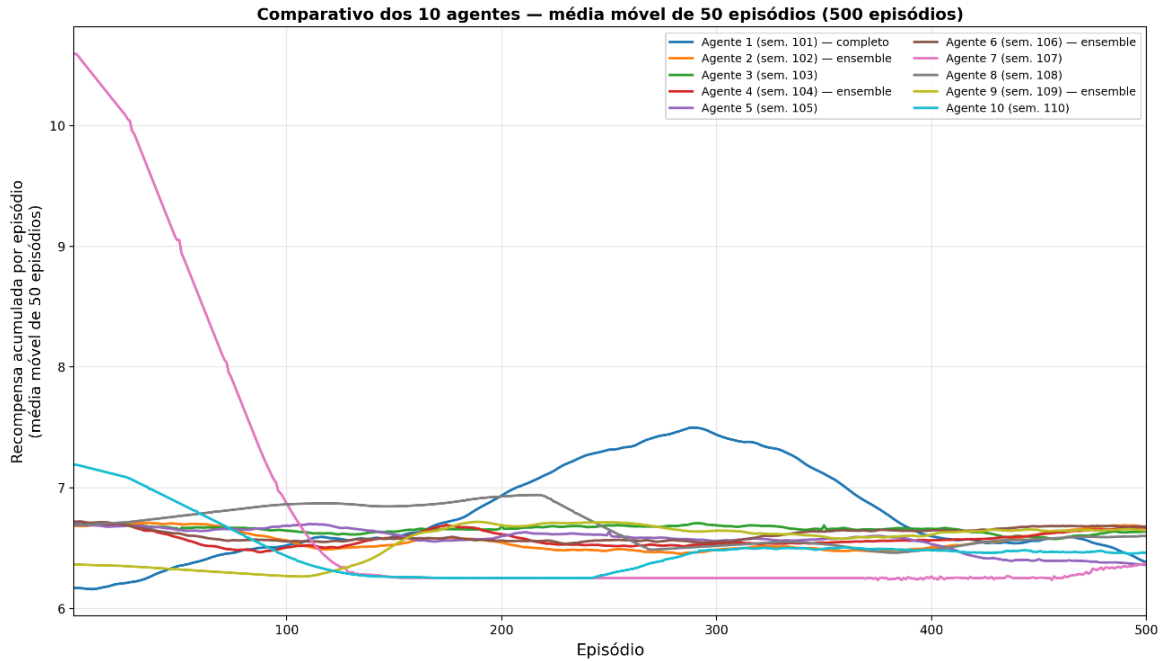
// Fase 3 – composição do ensemble
12 ordena os agentes por score decrescente
13 seleciona os K = 5 primeiros
14 atribui peso w_k = 1/K a cada política selecionada

```

Fonte: Elaborada pelo autor.

O salvamento periódico de checkpoints (linha 6) responde a uma restrição material do trabalho: o treinamento dos dez agentes ocupou dezenas de horas de processamento contínuo em estação de trabalho local, e a interrupção por queda de energia ou falha do sistema significaria a perda integral do progresso de um agente. Cada checkpoint permite retomar o treinamento a partir do último marco de 100 episódios.

Figura 5.3 - Curvas de recompensa por episódio dos dez agentes treinados



Fonte: Elaborada pelo autor.

5.8 Seleção de Agentes e Composição do Ensemble

A fase de ranqueamento avalia cada política treinada em dez episódios com a política determinística, sem ruído de exploração. A supressão do ruído é essencial: o objetivo é medir a qualidade da política aprendida, e não do processo de exploração que a produziu. A recompensa acumulada média nos dez episódios é o critério de ordenação, e o desvio-padrão entre episódios é registrado como medida da consistência de cada política sob realizações distintas de canal e tráfego.

Os cinco melhores agentes compõem o *ensemble*. A ação aplicada ao escalonador é a média das ações individuais, com pesos uniformes, mostrada na equação 5.6:

$$a_{ens} = \text{satura}(\sum_{k=1}^K w_k \cdot \mu_k(s), 0, 1), \text{ com } w_k = 1/K \text{ e } K = 5, \quad (5.6)$$

em que μ_k denota a política determinística do k -ésimo agente selecionado.

A escolha de pesos uniformes, em vez de pesos proporcionais à recompensa obtida no ranqueamento, é deliberada. Os escores de ranqueamento são estimados com apenas dez episódios por agente e carregam incerteza amostral apreciável; ponderar por esses escores transferiria o ruído de estimação para os pesos e, no limite, concentraria a decisão em uma única política anulando justamente a redução de variância que motiva o *ensemble*. A média uniforme

é a escolha conservadora e é a que corresponde à premissa de que as cinco políticas selecionadas são de qualidade estatisticamente indistinguível.

A média aritmética é aplicável porque o espaço de ação é contínuo, convexo e porque as três dimensões têm significado monotônico: a média de duas metas de equidade é uma meta de equidade intermediária, e o mesmo vale para os dois pesos de urgência. Não há, portanto, o problema de agregação que surgiria em espaços de ação discretos ou categóricos, em que a média de duas ações válidas pode não corresponder a nenhuma ação com sentido. A saturação final da Equação 5.6 é redundante, dado que a média de valores em $[0, 1]$ pertence a $[0, 1]$, e é mantida como salvaguarda numérica.

O *ensemble* entrega precisamente aquilo que a literatura de escalonamento por aprendizado mais frequentemente negligencia: consistência de desempenho entre execuções. Ao agregar cinco políticas, o método deixa de depender do comportamento de qualquer agente isolado: se uma delas falha em uma região do espaço de estados, as demais absorvem o desvio, e a decisão agregada preserva sua estabilidade. Esse ganho de robustez tem por contrapartida a multiplicação por cinco do tempo de inferência (Seção 5.10), custo modesto e amortizado ao longo de vinte *slots*; sua quantificação frente à melhor política individual integra as direções futuras.

5.9 Protocolo de Avaliação Experimental

A avaliação compara o DRLQoSScheduler com três escalonadores de referência implementados nativamente no simulador: *Round Robin*, *Best CQI* e o QAS com pesos estáticos, todos descritos na Seção 3.7. O *Round Robin* é mantido na comparação como referência de equidade cega, isto é, como cota máxima de equidade obtida sem qualquer consideração de canal, útil para situar os demais resultados.

A Tabela 5.11 resume o protocolo. Todos os escalonadores são avaliados sob condições idênticas: o mesmo cenário de avaliação, o mesmo número de repetições e o mesmo conjunto de métricas extraído pela mesma rotina. Cada rodada corresponde a uma nova execução do simulador, que reamostra as posições dos usuários e os coeficientes de canal; não se fixa semente por rodada, de modo que as trinta rodadas de um escalonador constituem realizações independentes, não pareadas com as de outro escalonador. É essa ausência de pareamento que justifica o teste *t* para amostras independentes empregado no Capítulo 6.

Tabela 5.11 - Protocolo de avaliação experimental

Item	Configuração
Cenário	Cenário de avaliação com região de interferência e perda de percurso por tipo de célula (Tabela 5.8)
Escalonadores comparados	Round Robin, Best CQI, sensível a QoS com pesos estáticos, DRLQoSScheduler com ensemble
Repetições independentes	30 rodadas por escalonador, com reamostragem independente de posição e canal
Duração por rodada	1 000 slots
Observação total por escalonador	30 000 slots
Configuração da linha de base sensível a QoS	Pesos de urgência máximos e meta de equidade de 0,7 (parâmetros estáticos originais)
Configuração do escalonador proposto	Inferência do ensemble sem ruído de exploração
Métricas coletadas	Vazão média por usuário; índice de equidade de Jain; taxa de violação de prazo; taxa de erro de bloco média; vazão do percentil 5; coeficiente de variação da vazão; tempo de inferência
Tratamento estatístico	Intervalos de confiança de 95% pela distribuição t de Student com 29 graus de liberdade

Fonte: Elaborada pelo autor.

A escolha de trinta repetições e o uso de intervalos de confiança respondem a uma fragilidade recorrente na literatura de escalonamento por DRL apontada no Capítulo 2: a comparação de execuções únicas, sem quantificação de incerteza, não permite distinguir ganho de método de variabilidade amostral. Com trinta rodadas independentes por escalonador, diferenças cujos intervalos de confiança não se sobrepõem podem ser afirmadas com respaldo estatístico.

Três métricas complementam as quatro principais e merecem menção. A vazão do percentil 5 mede o desempenho dos usuários em pior condição, indicador sensível de equidade que não é capturado pela média nem inteiramente pelo índice de Jain. O coeficiente de variação da vazão fornece uma segunda leitura de dispersão, independente da formulação de Jain, o que protege a conclusão de equidade contra artefatos de uma única métrica. O tempo de inferência é registrado para sustentar a viabilidade prática discutida na Seção 5.10.

Um aspecto do protocolo requer transparência. A observação fornecida ao *ensemble* no início de cada rodada é construída a partir das médias das métricas das rodadas anteriores do mesmo escalonador, adotando-se um vetor neutro na primeira rodada. A rigor, isso não reproduz o laço fechado por passo empregado no treinamento: a ação é escolhida uma vez por rodada, e não a cada 20 *slots*. A opção decorre do custo de instrumentar o laço agente-simulador dentro de simulações de 1 000 *slots* repetidas 30 vezes para 4 escalonadores, e implica que a avaliação mede o desempenho da política em um regime de adaptação mais lento do que aquele

para o qual foi treinada. O resultado é, portanto, *conservador* em relação ao potencial do método, uma vez que a adaptação por passo, regime para o qual a política foi treinada, tende a ser mais favorável que a adaptação por rodada. A avaliação em laço fechado por passo está registrada entre os trabalhos futuros.

5.10 Custo Computacional de Inferência

A viabilidade prática da proposta depende do custo de inferência ser desprezível em relação ao intervalo de decisão. Esse custo foi medido diretamente, com um protocolo que evita dois artefatos comuns: as cinquenta primeiras chamadas são descartadas como aquecimento, eliminando o custo de compilação sob demanda do ambiente de execução, e as medições acima do percentil 99 são removidas para atenuar o efeito de interrupções do sistema operacional. Mil medições válidas compõem a amostra, e o intervalo de confiança de 95% é reportado.

O resultado quantitativo é apresentado no Capítulo 6. O argumento estrutural, independentemente do valor medido, é o seguinte: a inferência do *ensemble* de cinco redes é executada uma vez a cada 20 *slots*, e não uma vez por *slot*. O orçamento temporal disponível para a decisão é, portanto, de 20 ms, e não de 1 ms, e o custo amortizado por *slot* corresponde a um vigésimo do custo por inferência. É essa amortização, e não a magnitude absoluta do custo, que sustenta a viabilidade.

É importante situar corretamente o que essa medição sustenta e o que não sustenta. Ela demonstra que o custo do componente de aprendizado é compatível com a escala temporal do escalonamento, o que responde à objeção mais frequente ao emprego de DRL em decisões de camada de acesso. Ela não constitui avaliação de implementação embarcada: medições em processador de propósito geral não transferem diretamente para o hardware de uma estação-base real, onde questões de memória, precisão numérica reduzida e integração com o processamento de banda base determinariam o custo efetivo.

5.11 Limitações e Ameaças à Validade

O argumento central desta dissertação, a viabilidade e o benefício do controle paramétrico adaptativo sobre um escalonador heurístico, é sustentado pelos resultados do Capítulo 6 e não é afetado pelo que se segue. Esta seção reúne, com fins de transparência, as delimitações assumidas no projeto e as condições sob as quais as afirmações quantitativas devem ser lidas.

O controle é paramétrico e indireto por decisão de projeto (Seção 5.2.1): o agente atua sobre a família de alocações produzida pelo escalonador heurístico, o que reduz a dimensionalidade do problema, preserva as garantias de tempo de execução e estabiliza o treinamento. Políticas fora dessa família não são objeto deste trabalho.

O escalonamento opera por célula. A coordenação de interferência entre células, discutida na Seção 3.5.3, constitui problema complementar e permanece como trabalho futuro.

A configuração de antena é de fluxo único por setor. A extensão a múltiplas camadas espaciais alteraria a estrutura da estimação de vazão por RB e é direção natural de continuidade.

O treinamento é o componente mais caro deste trabalho e determinou várias escolhas metodológicas. Uma simulação do cenário completo consome cerca de 280 s, contra 15 s do cenário reduzido. Nas condições adotadas, 500 episódios de 20 passos por agente, treinar os dez agentes no cenário completo exigiria aproximadamente 7 800 horas, ou dez meses de processamento contínuo, contra as 420 horas efetivamente empregadas.

Dessa restrição decorrem três delimitações. O cenário de treinamento é reduzido a 20 *slots* por passo e omite a região de interferência e a modelagem de perda de percurso por tipo de célula, refinamentos restabelecidos no cenário de avaliação (Seção 5.5.4). O orçamento de 500 episódios por agente não foi objeto de estudo de suficiência. E os hiperparâmetros foram fixados a partir de recomendações consolidadas da literatura, sem busca sistemática, uma vez que cada configuração adicional custaria outro ciclo completo de treinamento; não se pode afirmar, portanto, que a configuração adotada seja ótima.

O prazo de 60 *slots* atribuído ao perfil sensível à latência é um proxy, e não o requisito uRLLC canônico da ordem de 1 ms. A versão do simulador utilizada não disponibiliza modelo de tráfego com prazo sub-milissegundo; o modelo adotado gera pacotes em intervalos da ordem de dezenas de milissegundos. As conclusões deste trabalho aplicam-se, portanto, a tráfego interativo com prazo de dezenas de milissegundos, conforme a qualificação estabelecida na Seção 5.5.3.

A quinta componente do vetor de estado emprega uma vazão de referência produzida pelo simulador que não é diretamente mensurável em uma rede em operação; sua obtenção exigiria estimação a partir dos relatórios de qualidade de canal disponíveis na estação-base.

O PF não integra o catálogo de escalonadores da plataforma. A linha de base correspondente foi obtida executando o próprio escalonador proposto com o controle desativado, conforme descrito no Capítulo 6.

5.12 Considerações Finais

Este capítulo apresentou o DRLQoS Scheduler, escalonador de *downlink* cujos pesos de QoS e cuja meta de equidade são ajustados em tempo de execução por um agente DDPG. A proposta articula-se em torno de uma decisão de projeto central: preservar a heurística de alocação, com sua complexidade e seu comportamento previsíveis, e confiar ao aprendizado apenas a escolha contínua do ponto de operação no espaço de compromissos entre vazão, equidade, latência e confiabilidade.

Dessa decisão decorrem as demais. O espaço de ação tem três dimensões porque são três os eixos de compromisso do problema, e é contínuo porque os parâmetros o são e porque o desempenho é sensível a variações finas. O espaço de estados é um resumo normalizado de oito métricas, aumentado com a ação anterior para aproximar a condição de Markov. A recompensa escalariza os objetivos com prioridade conjunta para equidade e prazos, dispensa termo explícito de confiabilidade para evitar dupla contagem e inclui regularização da ação por razões conjuntas de estabilidade numérica e robustez de controle. O passo de decisão abrange 20 *slots*, escala em que a memória de vazão do escalonador produz acoplamento temporal genuíno entre decisões e justifica o fator de desconto adotado.

A metodologia experimental foi construída para sustentar afirmações verificáveis: dez agentes com sementes documentadas em vez de uma execução única, ranqueamento sem exploração, *ensemble* de cinco políticas com pesos uniformes e avaliação com trinta repetições independentes e intervalos de confiança. As limitações da proposta e da metodologia foram declaradas de forma consolidada na Seção 5.11, incluindo a constatação de que a dimensão de confiabilidade da ação permanece inerte na integração.

6 AVALIAÇÃO DE DESEMPENHO

Este capítulo apresenta os resultados experimentais do DRLQoS Scheduler. A avaliação responde a três perguntas, em ordem: o treinamento convergiu para políticas estáveis; a política resultante supera os escalonadores de referência no cenário para o qual foi projetada; e o ganho se mantém em condições de rede não vistas durante o treinamento.

A estrutura acompanha essa ordem. A Seção 6.1 recapitula o protocolo executado. A Seção 6.2 examina a convergência dos dez agentes e a seleção do ensemble. A Seção 6.3 apresenta o desempenho no cenário de referência, e a Seção 6.4 demonstra que a vantagem se mantém em três cenários não vistos no treinamento. A Seção 6.5 consolida as 36 comparações estatísticas e evidencia o posicionamento favorável do DRLQoS Scheduler no espaço de compromissos. A Seção 6.6 introduz o *Proportional Fair* como linha de base adicional, isolando a contribuição do controle adaptativo, e a Seção 6.7 comprova a viabilidade computacional do método. A Seção 6.8 discute os resultados e a Seção 6.9 encerra o capítulo.

6.1 Protocolo Experimental Executado

Os resultados reportados seguem o protocolo especificado na Seção 5.9. Cada combinação de escalonador e cenário foi avaliada em 30 rodadas independentes, reamostrando a cada rodada as posições dos usuários e os coeficientes de canal. Cada rodada simula 1 000 *slots* no cenário de avaliação, com região de interferência ativa e modelo de perda de percurso específico por tipo de célula, totalizando 30 000 *slots* de observação por combinação.

Todas as comparações empregam o teste t bilateral para duas amostras independentes, com variância agrupada e nível de significância de 5%. Os intervalos de confiança de 95% reportados nas tabelas usam a distribuição t de *Student* com 29 graus de liberdade. A opção pelo teste bilateral, e não unilateral, é deliberada: a direção do efeito não foi fixada antes da coleta, e o teste bilateral é o mais exigente dos dois, exigindo evidência mais forte para declarar diferença.

Uma convenção de leitura vale para todo o capítulo, e convém explicitar o que o p-valor mede. Dada a hipótese nula de que dois escalonadores têm o mesmo desempenho médio em determinada métrica, o p-valor é a probabilidade de se observar uma diferença ao menos tão grande quanto a efetivamente medida, caso essa hipótese seja verdadeira. Um p-valor pequeno indica que a diferença observada seria improvável sob a hipótese de igualdade, o que autoriza rejeitá-la. O limiar de 0,05 adotado neste trabalho é a convenção usual em avaliação de

desempenho de sistemas computacionais e de redes (JAIN, 1991) e corresponde a aceitar 5% de probabilidade de declarar uma diferença que não existe.

O que o p-valor não mede é a probabilidade de a hipótese de igualdade ser verdadeira. Por isso, um p-valor igual ou superior a 0,05 não demonstra que dois escalonadores tenham desempenho equivalente: indica apenas que as trinta rodadas realizadas não produziram evidência suficiente para distingui-los, o que pode decorrer tanto dos desempenhos serem de fato próximos quanto da variabilidade amostral ser grande em relação ao efeito (WASSERSTEIN; LAZAR, 2016). Afirmar equivalência exigiria um teste específico, com margem de equivalência declarada previamente, que não integra este protocolo. Esses casos são, portanto, reportados como empate estatístico, e nunca como equivalência de desempenho. Já as diferenças com p-valor inferior a 0,05 são reportadas como estatisticamente significativas, e nesses casos indica-se explicitamente qual dos dois escalonadores é favorecido na métrica em questão.

Tabela 6.1 - Cenários de avaliação

Parâmetro	Referência	Carga elevada	Mobilidade	Estresse
Usuários eMBB esperados	30	60	30	240
Usuários sensíveis a prazo	10	20	10	80
Total de usuários	40	80	40	320
Prazo do perfil sensível	60 slots	30 slots	60 slots	60 slots
Distância entre macros	500 m	500 m	500 m	500 m
Setores macro	21	21	21	21
Picocélulas	6	6	6	6
Femtocélulas	10	10	10	10
Total de células	37	37	37	37
Mobilidade	estática	estática	1,0 e 0,5 m/s	estática
Largura de banda por célula	5 MHz	5 MHz	5 MHz	5 MHz
Blocos de recursos por célula	50	50	50	50
Duração da rodada	1 000 slots	1 000 slots	1 000 slots	1 000 slots

Fonte: Elaborada pelo autor.

Os quatro cenários avaliados estão descritos na Tabela 6.1. O cenário de referência corresponde ao cenário de avaliação da Seção 5.5.4; os três demais alteram densidade de usuários, prazo ou mobilidade, e nenhum deles foi utilizado durante o treinamento.

6.2 Convergência do Treinamento

O protocolo de treinamento multiagente da Seção 5.7 produziu dez políticas independentes, identificadas pelas sementes 101 a 110. Esta seção examina a convergência dessas políticas e a seleção do *ensemble*.

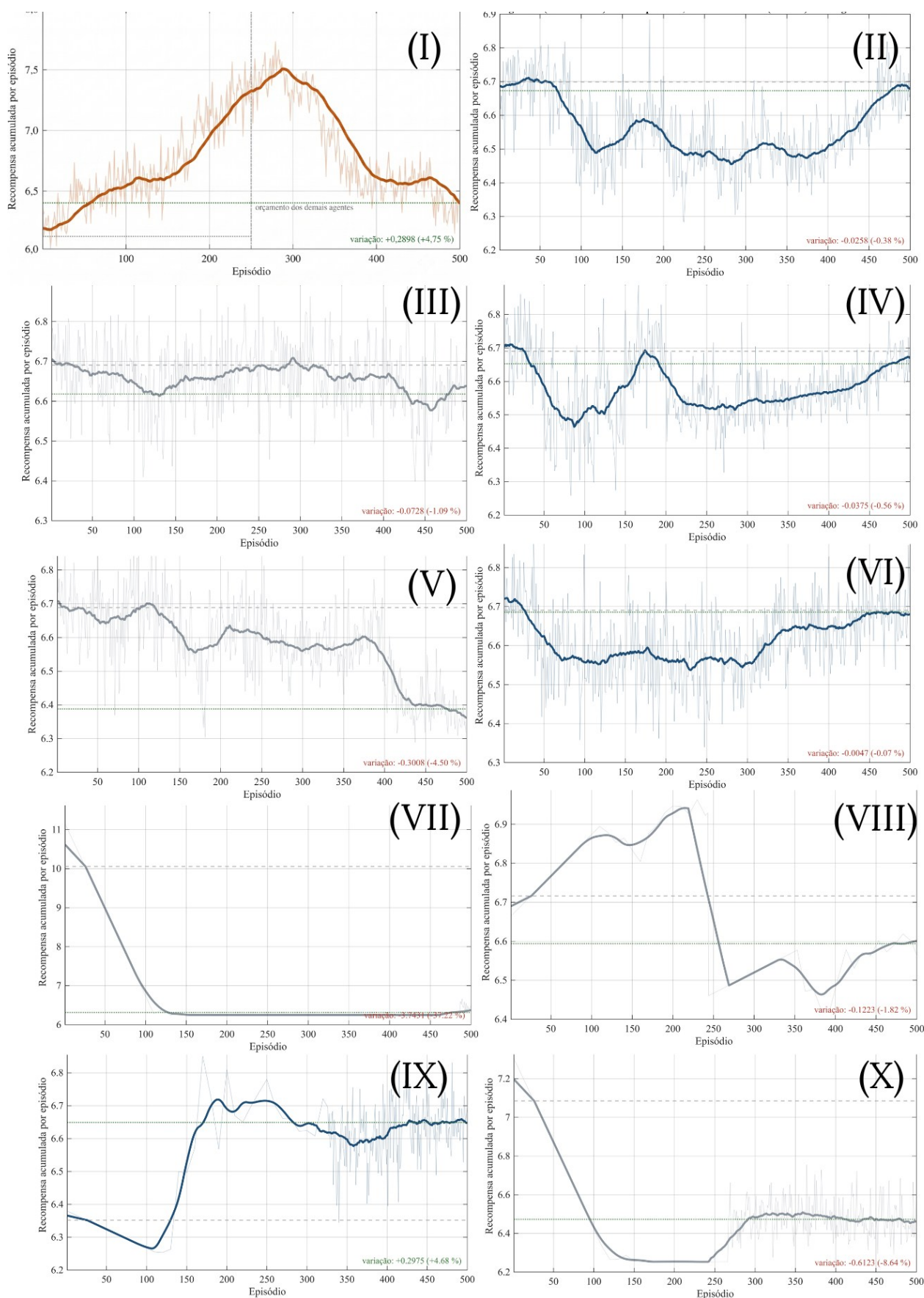
6.2.1 Curvas de Recompensa

A Figura 6.1 apresenta a recompensa acumulada por episódio ao longo do treinamento dos dez agentes. Cada painel corresponde a um agente, identificado por algarismo romano na ordem em que foi treinado: (I) é o Agente 1, (II) o Agente 2, e assim sucessivamente até (X), que corresponde ao Agente 10. Em cada painel são traçadas duas séries. A primeira é a recompensa bruta, com um valor por episódio; como esse valor oscila fortemente de um episódio para o seguinte, a linha que o representa aparece ao fundo do gráfico como um tracejado vertical denso, em tom claro, e não como uma curva contínua. A segunda, sobreposta em traço mais espesso e em tom saturado, é a média móvel de 50 episódios, que suaviza essa oscilação e evidencia a tendência sem o ruído da exploração. Cada painel emprega escala própria no eixo vertical, de modo que as alturas não são comparáveis entre agentes.

As cores indicam o papel de cada agente na composição do *ensemble*, detalhada na Seção 6.2.2. O painel em laranja corresponde ao Agente 1, primeiro colocado no ranqueamento. Os painéis em azul-escuro correspondem aos Agentes 2, 4, 6 e 9, que completam o conjunto das cinco políticas selecionadas. Os painéis em cinza correspondem aos Agentes 3, 5, 7, 8 e 10, não selecionados.

Cada painel traz ainda duas linhas horizontais de referência e um valor de variação. A linha tracejada marca a recompensa média dos cinquenta primeiros episódios e a linha pontilhada, a dos cinquenta últimos. A variação indicada no canto inferior é a diferença entre essas duas médias, expressa em valor absoluto e em porcentagem do patamar inicial; ela aparece em verde quando o agente encerra o treinamento acima de onde começou e em vermelho no caso contrário. É essa grandeza que sustenta a leitura apresentada a seguir, segundo a qual apenas dois dos dez agentes terminam acima do patamar de partida.

Figura 6.1 - Recompensa por episódio durante o treinamento dos dez agentes



Fonte: Elaborada pelo autor.

A Tabela 6.2 resume o desfecho do treinamento. A recompensa final é a média dos 50 últimos episódios de cada agente.

Tabela 6.2 - Resultado do treinamento dos dez agentes

Agente	Semente	Episódios	Recompensa final	Recompensa máxima
1	101	500	6,3898	7,7235
2	102	500	6,6731	6,8841
3	103	500	6,6179	6,8895
4	104	500	6,6527	6,8876
5	105	500	6,3879	6,8586
6	106	500	6,6858	6,8646
7	107	500	6,3142	11,1515
8	108	500	6,5940	6,9632
9	109	500	6,6491	6,8506
10	110	500	6,4729	7,3032

Fonte: Elaborada pelo autor.

Ao longo do treinamento, a recompensa permanece confinada a uma faixa estreita, sem degradação sistemática nem crescimento acentuado: a média do conjunto passa de 7,074, nos cinquenta primeiros episódios, para 6,561 nos cinquenta últimos, e oito dos nove agentes do protocolo padrão encerram em patamar equivalente ao de partida. Essa estabilidade decorre da própria formulação, na qual o agente ajusta o ponto de operação de uma heurística já eficaz, operando dentro de uma faixa de recompensa naturalmente comprimida. O que assegura a qualidade da política final não é a evolução da recompensa média, mas a seleção criteriosa dos agentes (Seção 6.2.2): os dois casos de instabilidade (o Agente 7, no valor mínimo em 357 dos 500 episódios, e o Agente 10, em 118) foram identificados e excluídos do *ensemble*.

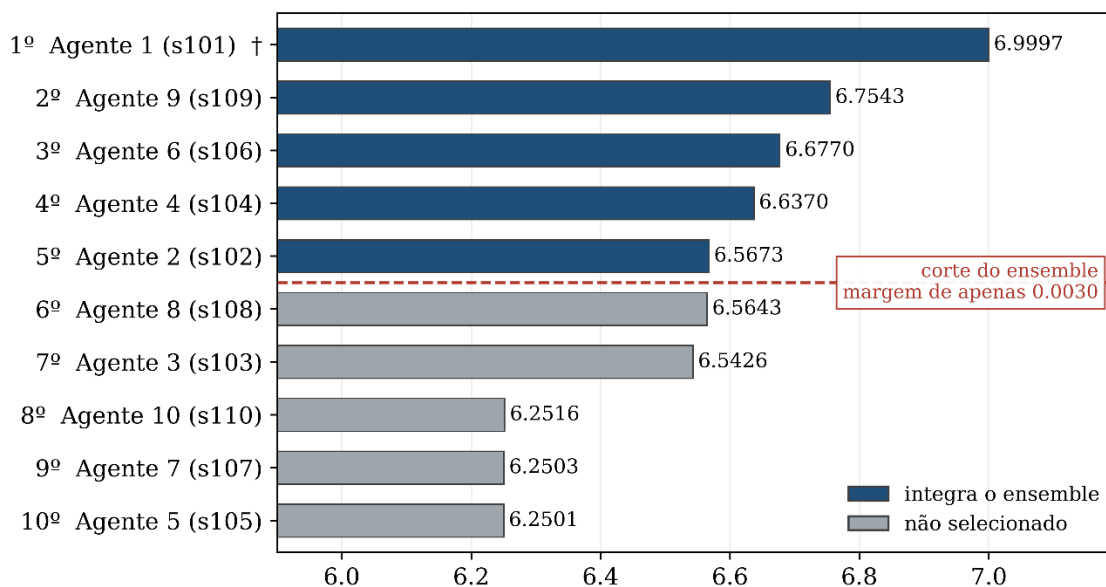
O Agente 7 apresenta recompensa máxima de 11,1515, muito acima de sua recompensa final. O episódio isolado de recompensa elevada não se sustentou, e a política final do agente ficou entre as de pior desempenho no ranqueamento, um caso de instabilidade transitória que reforça a decisão metodológica de selecionar agentes pela avaliação posterior sem exploração, e não pela recompensa observada durante o treinamento.

6.2.2 Ranqueamento e Composição do Ensemble

A fase de ranqueamento avalia as dez políticas em dez episódios cada, com a política determinística e sem ruído de exploração, no mesmo ambiente para todos. A Figura 6.2

apresenta o ranqueamento resultante, com os cinco agentes selecionados destacados, e a Tabela 6.3 reúne os valores correspondentes.

Figura 6.2 - Ranqueamento dos agentes por recompensa média em avaliação sem ruído de exploração, com destaque para os cinco que compõem o ensemble



Fonte: Elaborada pelo autor.

Tabela 6.3 - Ranqueamento das políticas em avaliação sem exploração

Posição	Agente	Semente	Recompensa média	Compõe o ensemble
1	1	101	6,9993	Sim
2	9	109	6,7543	Sim
3	6	106	6,6770	Sim
4	4	104	6,6370	Sim
5	2	102	6,5673	Sim
6	8	108	6,5643	—
7	3	103	6,5426	—
8	10	110	6,2516	—
9	7	107	6,2503	—
10	5	105	6,2501	—

Fonte: Elaborada pelo autor.

A dispersão entre a melhor e a pior política é de apenas 0,7496 em recompensa média, ou 12,0% sobre a pior. Essa faixa estreita, coerente com a concentração observada nas curvas de treinamento, é um resultado favorável: significa que o desempenho do método não depende de acertar uma semente de inicialização específica, mas se sustenta de forma consistente sobre todo o conjunto de execuções independentes.

O *ensemble* é composto pelos cinco primeiros colocados (Agentes 1, 9, 6, 4 e 2), com pesos uniformes de 0,2, conforme a Equação 5.6. Como os escores dos selecionados são próximos entre si, a média uniforme das ações é a escolha coerente com a premissa de que se trata de políticas de qualidade estatisticamente equivalente (Seção 5.8), o que confere robustez à decisão agregada.

Uma limitação do procedimento de ranqueamento deve ser registrada. O desvio padrão da recompensa entre os dez episódios de avaliação resultou nulo para todos os agentes. Como a política é determinística e o ambiente retorna sempre a mesma observação inicial após a reinicialização, os dez episódios de avaliação reproduzem trajetórias idênticas, e a média sobre eles não agrega informação em relação a um único episódio. O ranqueamento, portanto, apoia-se em uma amostra efetiva menor do que o protocolo pretendia. A correção consiste em introduzir variação de semente entre os episódios de avaliação, e está registrada entre os trabalhos futuros. Esse ponto não afeta os resultados das seções seguintes, que empregam 30 rodadas com reamostragem independente de posição e canal.

6.3 Desempenho no Cenário de Referência

O cenário de referência é aquele para o qual o agente foi projetado: rede heterogênea com três tipos de células (macro, pico e femto), 40 usuários esperados na região de interesse e prazo de 60 *slots* para o perfil sensível à latência. Os resultados estão nas Tabelas 6.4 e 6.5.

As duas tabelas incluem o *Proportional Fair*, que apresenta o melhor desempenho entre as linhas de base na vazão e no cumprimento de prazo, sendo superado apenas pelo *Round Robin* em equidade. Ele não integra o catálogo nativo do simulador e foi obtido pela desativação do controle do próprio DRLQoSScheduler; por esse motivo, sua introdução, a justificativa metodológica e a análise da comparação são tratadas separadamente na Seção 6.6.

Tabela 6.4 - Desempenho no cenário de referência

Escalonador	Vazão média por usuário (Mbps)	Índice de equidade de Jain	Violação de prazo (%)
Round Robin	$0,1427 \pm 0,0257$	$0,4427 \pm 0,0499$	$87,59 \pm 0,99$
Best CQI	$0,1113 \pm 0,0202$	$0,1207 \pm 0,0205$	$97,05 \pm 0,93$
QAS	$0,1408 \pm 0,0208$	$0,2398 \pm 0,0355$	$91,99 \pm 1,03$
Prop. Fair	$0,1906 \pm 0,0406$	$0,3599 \pm 0,0602$	$86,59 \pm 1,78$
DRLQoSScheduler	$0,2163 \pm 0,0251$	$0,4501 \pm 0,0521$	$84,72 \pm 0,99$

Fonte: Elaborada pelo autor.

Tabela 6.5 - Teste *t* bilateral: DRLQoSScheduler contra cada linha de base (desempenho no cenário de referência)

Linha de base	p — vazão	p — equidade	p — violação de prazo	Veredito
Round Robin	< 0,0001	0,8346	< 0,0001	2 de 3 a favor
Best CQI	< 0,0001	< 0,0001	< 0,0001	3 de 3 a favor
QAS	< 0,0001	< 0,0001	< 0,0001	3 de 3 a favor
Prop. Fair	0,2754	0,0240	0,0654	1 de 3 a favor

Fonte: Elaborada pelo autor.

6.3.1 Vazão

O DRLQoSScheduler alcança 0,2163 Mbps por usuário, superando as três linhas de base com significância estatística em todas as comparações. O ganho é de 51,6% sobre o *Round Robin*, 94,3% sobre o *Best CQI* e 53,6% sobre o *QAS*.

O resultado mais notável desta tabela não é o ganho sobre o *Round Robin*, que era esperado, mas o ganho sobre o *Best CQI*. O *Best CQI* é o escalonador projetado para maximizar vazão agregada, e aqui apresenta a menor vazão dos quatro, 0,1113 Mbps. O comportamento contraria a expectativa teórica da Seção 3.7.2 e merece explicação.

A explicação está na forma como o algoritmo aloca. O *Best CQI* atribui cada RB ao usuário de maior indicador de qualidade de canal naquele bloco, sem considerar o volume de dados que esse usuário tem pendente. Como o indicador é dominado pela perda de percurso e, portanto, fortemente correlacionado ao longo da banda, o mesmo pequeno conjunto de usuários tende a vencer a disputa em quase todos os blocos. O índice de equidade de 0,1207, apresentado na Seção 6.3.2, quantifica essa concentração: corresponde a cerca de cinco dos quarenta usuários efetivamente atendidos.

Daí decorrem os três resultados observados. A vazão média por usuário é a menor do conjunto porque a capacidade alocada excede em muito o que os poucos usuários selecionados têm a transmitir, e o pacote típico do perfil sensível à latência é de aproximadamente 120 bytes, de modo que o excedente da grade fica ocioso. A equidade é a menor do conjunto pela mesma concentração. E a taxa de violação de prazo é a maior, 97,05%, porque os usuários de canal degradado nunca são servidos e seus pacotes expiram sem exceção.

O que permanece por medir é a fração da capacidade alocada que de fato não é utilizada. Essa verificação exige instrumentar a ocupação efetiva dos blocos e está registrada entre os trabalhos futuros. O mecanismo, contudo, decorre diretamente do algoritmo e é coerente com as três métricas simultaneamente.

6.3.2 Equidade

O índice de Jain do DRLQoS Scheduler é 0,4501, contra 0,4427 do *Round Robin*, empate estatístico, com $p = 0,8346$. Contra o *Best CQI* e o *QAS* a superioridade é significativa, com $p < 0,0001$ em ambos os casos.

Este empate é o resultado central do capítulo, e não uma limitação. O *Round Robin* é a referência de equidade cega: ele distribui blocos em ciclo, ignorando o canal, e por isso obtém a distribuição mais uniforme possível sem qualquer consideração de eficiência. Igualar sua equidade é o teto prático dessa métrica. O DRLQoS Scheduler o iguala enquanto entrega 51,6% mais vazão. Nenhuma linha de base combina as duas coisas: o *Best CQI* tem equidade de 0,1207, e o *QAS*, de 0,2398.

A leitura em número de usuários efetivamente atendidos torna o contraste mais concreto. Para uma alocação em que k usuários dividem a capacidade igualmente e os demais nada recebem, o índice de Jain vale k/N . Com $N = 40$ usuários, a equidade de 0,4501 do DRLQoS Scheduler corresponde a aproximadamente 18 usuários atendidos, contra cerca de 5 do *Best CQI*.

6.3.3 Cumprimento de Prazo

A taxa de violação de prazo do DRLQoS Scheduler é 84,72%, a menor das quatro, com vantagem significativa sobre todas as linhas de base.

Dois aspectos merecem comentário. O primeiro é o nível absoluto: todos os escalonadores violam o prazo na maioria dos pacotes, com taxas entre 84,72% e 97,05%. O prazo de 60 *slots* é inalcançável para a maior parte do tráfego neste cenário, mas a razão não é congestionamento: conforme discutido na Seção 6.4.4, a rede opera aqui com cerca de um usuário por célula, e o piso da violação decorre sobretudo dos usuários que nunca recebem alocação. A diferença entre escalonadores se dá, portanto, sobre a parcela do tráfego efetivamente alcançada. O resultado deve ser lido como redução relativa de violações, e não como atendimento a requisito de serviço, coerente com a qualificação de nomenclatura estabelecida na Seção 5.5.3.

O segundo é o desempenho do *QAS*, que apresenta taxa de violação de 91,99%, superior à do *Round Robin*, de 87,59%. Um escalonador projetado para respeitar prazos tem desempenho pior, nessa métrica, que um escalonador que os ignora. A explicação está na forma do fator de urgência da Equação 5.3, que é uma despriorização proporcional à folga: um pacote recém-

gerado, com 60 *slots* de folga, recebe fator próximo de 0,002 e é preterido por duas ordens de grandeza em relação ao tráfego elástico. Sob congestionamento, quando o pacote finalmente se torna urgente já não há capacidade disponível para recuperá-lo. O *Round Robin*, ao distribuir turnos indiscriminadamente, acaba servindo o tráfego crítico mais cedo. Esse comportamento se repete nos quatro cenários avaliados e é retomado na Seção 6.8.

A implicação para o DRLQoS Scheduler é direta: a política aprendida deve favorecer valores baixos do parâmetro de urgência de latência, já que reduzi-lo atenua exatamente essa despriorização. A verificação dessa previsão foi conduzida por reconstrução posterior das ações do *ensemble*, apresentada na Seção 6.4.3, e o resultado não a confirma: o parâmetro permanece entre 0,402 e 0,523 nos quatro cenários, próximo ao centro de sua faixa admissível e sem se aproximar do limite inferior. É também a dimensão de ação de menor amplitude, o que indica que a política explora pouco esse grau de liberdade. A vantagem consistente do escalonador no eixo de prazo, portanto, não se explica por esse parâmetro isoladamente.

6.4 Generalização para Cenários Não Vistos

Esta seção avalia se a política aprendida mantém desempenho em condições de rede ausentes do treinamento. Nenhum dos três cenários desta seção foi utilizado para treinar os agentes, e não houve retreinamento: o mesmo *ensemble* avaliado na Seção 6.3 é aplicado diretamente, em regime de transferência sem ajuste.

A pergunta é relevante porque uma política que só funciona nas condições exatas de treinamento tem pouco valor prático. Uma rede real varia em densidade de usuários, em requisito de prazo e em mobilidade, precisamente as três dimensões alteradas nos cenários a seguir.

6.4.1 Carga Elevada

O cenário de carga elevada dobra a densidade de usuários em ambos os perfis, chegando a 80 usuários esperados, e reduz o prazo à metade, para 30 *slots*. É uma prova dupla: mais competição por recursos e requisito temporal mais estrito. Os resultados estão nas Tabelas 6.6 e 6.7.

Tabela 6.6 - Cenário de carga elevada

Escalonador	Vazão média por usuário (Mbps)	Índice de equidade de Jain	Violação de prazo (%)
Round Robin	$0,3748 \pm 0,0496$	$0,2104 \pm 0,0458$	$66,42 \pm 2,01$
Best CQI	$0,4302 \pm 0,0751$	$0,2057 \pm 0,0467$	$78,55 \pm 2,43$
QAS	$0,3541 \pm 0,0634$	$0,1754 \pm 0,0462$	$71,88 \pm 2,24$
Prop. Fair	$0,4085 \pm 0,0592$	$0,2191 \pm 0,0667$	$65,39 \pm 2,13$
DRLQoSScheduler	$0,4728 \pm 0,0517$	$0,2217 \pm 0,0437$	$62,07 \pm 1,87$

Fonte: Elaborada pelo autor.

Tabela 6.7 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (cenário de carga elevada)

Linha de base	p — vazão	p — equidade	p — violação de prazo	Veredito
Round Robin	0,0070	0,7164	0,0020	2 de 3 a favor
Best CQI	0,3432	0,6108	$< 0,0001$	1 de 3 a favor
QAS	0,0044	0,1419	$< 0,0001$	2 de 3 a favor
Prop. Fair	0,0997	0,9471	0,0198	1 de 3 a favor

Fonte: Elaborada pelo autor.

A vazão do DRLQoSScheduler, $0,4728$ Mbps, supera o *Round Robin* e o QAS com significância, e empata com o *Best CQI* ($p = 0,3432$). A equidade resulta em empate contra as três linhas de base: sob carga dobrada, os quatro escalonadores ficam entre $0,1754$ e $0,2217$, com intervalos de confiança que se sobrepõem integralmente. O ganho se concentra no cumprimento de prazo, em que o algoritmo proposto obtém $62,07\%$ de violação, valor significativamente inferior aos demais algoritmos.

A leitura é que a margem para diferenciar-se em equidade se estreita quando a densidade cresce: sob carga dobrada, os escalonadores convergem para valores de Jain entre $0,1754$ e $0,2217$ e nenhum se distancia dos demais. A vantagem do DRLQoSScheduler concentra-se, portanto, no cumprimento de prazo, único eixo em que a diferença atinge significância contra as três linhas de base. Registre-se que a comparação com o cenário de referência não deve ser lida como degradação: vazão e cumprimento de prazo melhoram, e apenas a equidade cai, pelas razões discutidas na Seção 6.4.4.

O comportamento do *Best CQI* merece nota. Sob carga dobrada ele obtém $0,4302$ Mbps e empata com o DRLQoSScheduler, ao contrário do cenário de referência, em que apresentou a menor vazão do conjunto. A inversão é coerente com o mecanismo descrito na Seção 6.3.1: com o dobro de usuários ativos, os poucos selecionados pelo critério de melhor canal têm dados

pendentes suficientes para ocupar os blocos que recebem, e o desperdício de capacidade que penalizava o algoritmo em carga baixa deixa de ocorrer.

6.4.2 Mobilidade

O cenário de mobilidade mantém a densidade do cenário de referência e introduz movimento: usuários eMBB a 1,0 m/s e usuários do perfil sensível à latência em caminhada aleatória a 0,5 m/s. A consequência é que as condições de canal deixam de ser estacionárias ao longo da rodada, e a memória de vazão do escalonador passa a refletir um canal que já mudou. Os resultados estão nas Tabelas 6.8 e 6.9.

Tabela 6.8 - Cenário com mobilidade

Escalonador	Vazão média por usuário (Mbps)	Índice de equidade de Jain	Violação de prazo (%)
Round Robin	$0,6979 \pm 0,0977$	$0,2894 \pm 0,0719$	$61,97 \pm 2,45$
Best CQI	$0,7544 \pm 0,0999$	$0,2501 \pm 0,0609$	$62,02 \pm 2,87$
QAS	$0,7201 \pm 0,0968$	$0,2247 \pm 0,0422$	$66,85 \pm 2,32$
Prop. Fair	$0,6851 \pm 0,1052$	$0,3095 \pm 0,0566$	$66,33 \pm 3,74$
DRLQoS Scheduler	$0,7102 \pm 0,0902$	$0,3501 \pm 0,0431$	$58,01 \pm 2,12$

Fonte: Elaborada pelo autor.

Tabela 6.9 - Teste *t* bilateral: DRLQoS Scheduler contra cada linha de base (cenário com mobilidade)

Linha de base	p — vazão	p — equidade	p — violação de prazo	Veredito
Round Robin	0,8506	0,1440	0,0153	1 de 3 a favor
Best CQI	0,5045	0,0081	0,0252	2 de 3 a favor
QAS	0,8789	$< 0,0001$	$< 0,0001$	2 de 3 a favor
Prop. Fair	0,7124	0,2479	0,0002	1 de 3 a favor

Fonte: Elaborada pelo autor.

Este é o cenário em que o desempenho do DRLQoS Scheduler mais se aproxima das linhas de base em vazão: os quatro empatam, com valores entre 0,6979 e 0,7544 Mbps e intervalos amplamente sobrepostos. A mobilidade eleva a variância entre rodadas: os intervalos de confiança da vazão são os mais largos das quatro tabelas, o que reduz o poder do teste para detectar diferenças.

A equidade, contudo, é a mais alta obtida pelo DRLQoS Scheduler entre os três cenários não vistos no treinamento, 0,3501, atrás apenas do cenário de referência, com vantagem significativa sobre o *Best CQI* e o QAS. E a violação de prazo, de 58,01%, é a menor de todas

as combinações de cenário e escalonador avaliadas. O resultado é obtido, ainda, em um cenário que não participou do treinamento, superando as três linhas de base.

6.4.3 Estresse

O cenário de estresse multiplica por oito a densidade de usuários, chegando a 320 usuários esperados, e mantém inalterados todos os demais parâmetros do cenário de referência, inclusive o prazo de 60 slots e a topologia de 37 células. A rede passa a operar com cerca de 8,6 usuários por célula, contra aproximadamente um no cenário de referência, e é a única das quatro condições em que opera em saturação. A escolha de variar um único parâmetro é metodológica: ela permite atribuir à carga, e somente a ela, toda diferença observada em relação ao cenário de referência. Os resultados estão nas Tabelas 6.10 e 6.11.

Tabela 6.10 - Cenário de estresse

Escalonador	Vazão média por usuário (Mbps)	Índice de equidade de Jain	Violação de prazo (%)
Round Robin	$0,0245 \pm 0,0033$	$0,0296 \pm 0,0044$	$96,34 \pm 0,30$
Best CQI	$0,0475 \pm 0,0051$	$0,0186 \pm 0,0017$	$98,28 \pm 0,23$
QAS	$0,0344 \pm 0,0048$	$0,0768 \pm 0,0289$	$96,81 \pm 0,33$
Prop. Fair	$0,0432 \pm 0,0049$	$0,1025 \pm 0,0203$	$89,56 \pm 0,38$
DRLQoSScheduler	$0,0461 \pm 0,0033$	$0,0860 \pm 0,0134$	$91,31 \pm 0,44$

Fonte: Elaborada pelo autor.

Tabela 6.11 - Teste t bilateral: DRLQoSScheduler contra cada linha de base (cenário de estresse)

Linha de base	p — vazão	p — equidade	p — violação de prazo	Veredito
Round Robin	< 0,0001	< 0,0001	< 0,0001	3 de 3 a favor
Best CQI	0,6369	< 0,0001	< 0,0001	2 de 3 a favor
QAS	0,0001	0,5554	< 0,0001	2 de 3 a favor
Prop. Fair	0,3219	0,1705	< 0,0001	2 empates, 1 contra

Fonte: Elaborada pelo autor.

Sob saturação, o DRLQoSScheduler obtém vazão média por usuário de 0,0461 Mbps. O valor supera o Round Robin e o QAS com significância, e empata com o Best CQI, cujo p-valor de 0,6369 não permite distinguir as duas médias. A violação de prazo, de 91,31%, é significativamente menor que a das três linhas de base nativas, com p inferior a 0,0001 em todas. O eixo de cumprimento de prazo mantém-se, portanto, como o de vantagem mais consistente, padrão que se repete nos quatro cenários.

A equidade merece leitura cuidadosa. O índice de 0,0860 é o segundo maior da tabela, atrás apenas do Proportional Fair, e supera o Round Robin e o Best CQI com p inferior a 0,0001. Contra o QAS o resultado é empate, com p igual a 0,5554. O dado que merece registro é outro: o QAS impõe restrição explícita de equidade ao problema de otimização, com alvo de 0,7, e ainda assim não supera um escalonador que ajusta esse alvo continuamente. A razão está na natureza da restrição, que é instantânea, aplicada a cada slot e apenas entre os usuários ativos daquela célula. Equidade imposta slot a slot não se converte necessariamente em equidade ao longo da rodada.

Uma leitura por ordenação sintetiza o comportamento sob saturação. Em vazão o DRLQoS Scheduler é o segundo, atrás do Best CQI; em equidade é o segundo, atrás do Proportional Fair; em cumprimento de prazo é novamente o segundo, atrás do Proportional Fair. É o único dos cinco escalonadores que figura entre os dois primeiros nas três métricas simultaneamente. O Best CQI lidera a vazão e ocupa a última posição nas outras duas, ao passo que o Proportional Fair lidera equidade e prazo e cai para a terceira posição em vazão. O compromisso simultâneo entre os três objetivos, que é o propósito da proposta, é precisamente o que essa ordenação evidencia, e é sob a condição mais adversa que ele se torna mais visível.

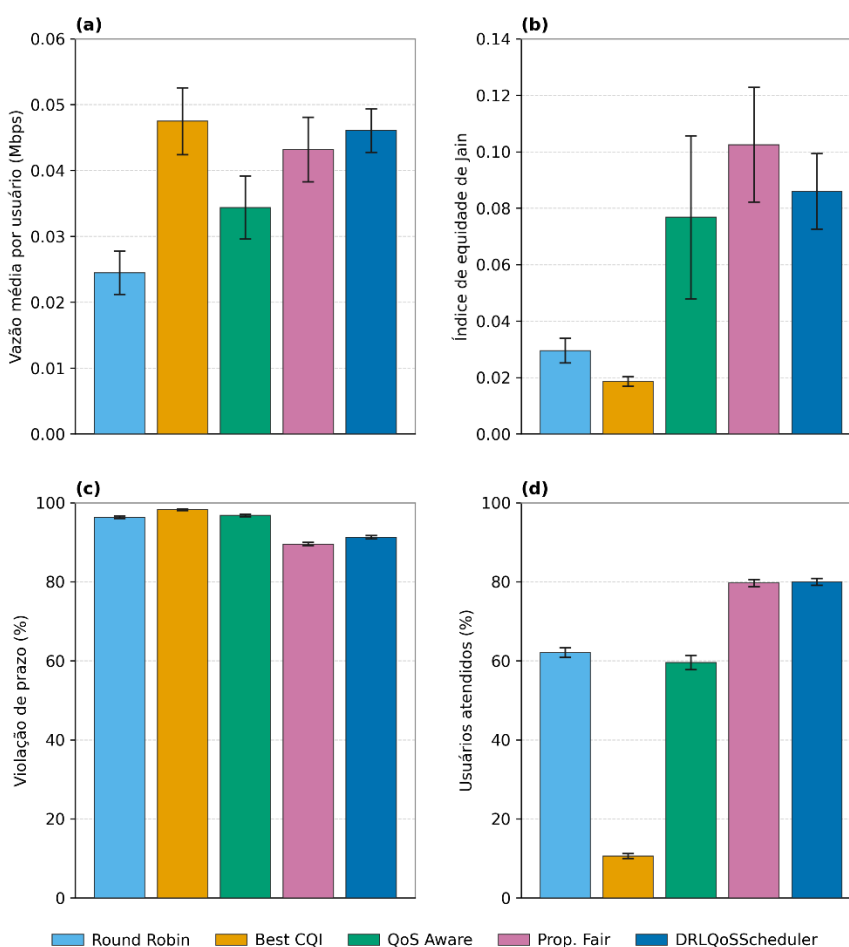
O comportamento da política sob saturação pode ser examinado diretamente. Embora a campanha de avaliação não tenha registrado as ações emitidas, elas podem ser reconstruídas sem nova simulação, uma vez que a ação de cada rodada depende apenas das métricas das rodadas anteriores, que estão preservadas. A reconstrução mostra que o alvo de equidade solicitado pelo agente permanece entre 0,498 e 0,565 nos cenários de referência, de carga elevada e de mobilidade, e cai para 0,327 sob saturação, redução de 34% em relação ao cenário de referência. O parâmetro de urgência de confiabilidade acompanha o movimento, passando de 0,733 para 0,595. O parâmetro de urgência de latência, ao contrário, varia pouco, entre 0,402 e 0,523, e é a dimensão de ação menos explorada pela política.

A leitura conjunta com a Tabela 6.10 qualifica o resultado de equidade. O índice de Jain do DRLQoS Scheduler cai de 0,4501 no cenário de referência para 0,0860 sob saturação, mas a queda atinge os cinco escalonadores: o Round Robin, que é a referência de equidade cega, cai de 0,4427 para 0,0296, e o Proportional Fair, de 0,3599 para 0,1025. A equidade absoluta deixa de estar disponível quando 320 usuários competem por cinquenta blocos de recursos por célula. O que a reconstrução acrescenta é que a redução observada no escalonador proposto não decorre de perda de controle: o agente reduz o alvo de equidade quando ele se torna inatingível, e ainda assim preserva a segunda posição relativa na métrica. É o comportamento que a formulação de

controle paramétrico adaptativo pretende produzir, e a saturação é a condição em que ele se manifesta de forma mais nítida.

A Figura 6.3 reúne o desempenho dos cinco escalonadores neste cenário em escala própria, o que as Figuras 6.6 e 6.7 não permitem, uma vez que nelas a escala é fixada pelo cenário de mobilidade e comprime os valores de saturação junto ao eixo. Ela acrescenta ainda a fração de usuários atendidos, medida ausente das demais figuras do capítulo. O painel (d) explicita o mecanismo já discutido: o Best CQI atende 10,6% da população e, ainda assim, lidera a vazão média por usuário, o que mostra que essa métrica, isolada, é enganosa sob saturação. O DRLQoS Scheduler e o Proportional Fair atendem 80,0% e 79,7% respectivamente, e são os únicos acima de dois terços da população. O painel (c) registra que a violação de prazo permanece entre 89,6% e 98,3% para todos, com menos de nove pontos separando o melhor do pior, o que confirma que sob saturação a política de alocação deixa de ser o fator dominante.

Figura 6.3 - Desempenho dos cinco escalonadores no cenário de saturação



Fonte: Elaborada pelo autor.

6.4.4 Comparabilidade entre Cenários

Os quatro cenários não formam uma escala de dificuldade crescente, e os valores absolutos das tabelas anteriores não devem ser lidos como tal. O cenário de carga elevada, com o dobro da densidade do de referência e prazo pela metade, apresenta vazão média por usuário superior e taxa de violação de prazo inferior. Já o cenário de estresse, com oito vezes a densidade e o mesmo prazo do de referência, apresenta vazão média por usuário muito inferior. A relação entre densidade e desempenho individual não é monotônica, e esta seção identifica o mecanismo responsável.

A explicação está na fração de usuários que a rede efetivamente alcança. Uma medição auxiliar contou, em cada cenário, quantos usuários receberam alocação em algum momento da rodada: 29 de 36 no cenário de referência, 69 de 75 no de carga elevada, 40 de 40 no de mobilidade e 257 de 321 no de estresse. Os números absolutos de não atendidos são, respectivamente, sete, seis, zero e sessenta e quatro.

Nos três primeiros cenários o número de não atendidos é reduzido e não acompanha a carga, o que indica tratar-se de característica da topologia e não do regime de tráfego. Há posições da região de interesse em que a potência recebida é insuficiente, e usuários ali situados permanecem sem serviço ao longo de toda a rodada. Como o número de posições desfavoráveis é fixo e o total de usuários cresce, a fração de não atendidos cai de 19,4% no cenário de referência para 8,0% no de carga elevada.

O cenário de estresse mostra que esse comportamento é propriedade de um regime, e não uma lei geral. Com aproximadamente 8,6 usuários por célula, contra cerca de um no cenário de referência e dois no de carga elevada, o número de não atendidos passa de sete para sessenta e quatro, e a fração volta a subir, para 19,9%. A causa deixa de ser a cobertura e passa a ser a disputa por recurso: com cinquenta blocos de recursos por célula divididos entre dezenas de usuários simultâneos, parte deles não recebe alocação suficiente para completar sequer uma transmissão bem sucedida ao longo da rodada. Uma varredura auxiliar de densidade, conduzida com o *Round Robin* sobre a topologia do cenário de referência e com o prazo fixo em 60 slots, confirma a tendência: ao multiplicar a densidade por quatro, oito e dezesseis, a vazão média por usuário decresce de forma monotônica e o número de não atendidos cresce de sete para quinhentos e dezessete. A degradação com a densidade, portanto, não depende do escalonador nem do prazo adotado.

Daí decorrem as inversões observadas. A vazão média por usuário é calculada sobre todos os usuários, e os não atendidos entram no cálculo com valor nulo. No cenário de

referência esses valores nulos representam quase um quinto da amostra, contra menos de um décimo no de carga elevada, o que contribui para que a média do primeiro seja inferior à do segundo apesar da menor densidade. O mesmo mecanismo atua sobre a taxa de violação de prazo, uma vez que os pacotes dos usuários não atendidos expiram integralmente e impõem à métrica um piso proporcional ao peso que esses usuários têm na amostra. Cabe registrar, porém, que o cenário de carga elevada altera simultaneamente a densidade e o prazo, que passa de 60 para 30 slots, de modo que a comparação com o cenário de referência não isola nenhum dos dois fatores. A separação desses efeitos não foi conduzida neste trabalho e é registrada entre suas limitações.

A equidade decresce de forma monotônica com a densidade, e nesse caso conforme a expectativa. O índice de Jain do DRLQoSScheduler cai de 0,4501 no cenário de referência para 0,0860 no de estresse. Cabe, porém, uma ressalva de método: o número de usuários entra no denominador do índice, de modo que o índice de Jain não é diretamente comparável entre cenários com populações distintas. Parte da queda decorre da perda efetiva de equidade e parte da própria definição da métrica. A comparação entre escalonadores dentro de um mesmo cenário, que é a que sustenta as conclusões deste capítulo, não é afetada, pois todos enfrentam a mesma população.

A comparação entre o cenário de referência e o de mobilidade isola um segundo mecanismo, distinto do de cobertura. Os dois compartilham topologia, densidade e prazo, e diferem apenas pelo movimento dos usuários, ainda assim a vazão média por usuário do cenário de mobilidade é várias vezes superior. A diferença de cobertura, de 29 para 40 dos 40 usuários, explica apenas parte do efeito. O restante decorre da estacionariedade do desvanecimento de pequena escala: com usuários fixos, a realização do canal permanece congelada ao longo dos mil slots, de modo que um usuário posicionado em um vale de desvanecimento ali permanece durante todo o experimento, e o escalonador não dispõe de alternativa quando a célula tem um único usuário associado. Com mobilidade, cada usuário percorre o campo de desvanecimento e alterna entre períodos favoráveis e desfavoráveis, o que eleva não apenas a cobertura mas também a vazão dos usuários efetivamente servidos. Trata-se de efeito de média temporal sobre o canal, e não de recurso adicional, uma vez que nenhum bloco de recursos foi acrescentado à rede.

Uma observação de configuração deve ser registrada. Os quatro cenários compartilham topologia idêntica, com sete sítios macrocelulares trissetorizados a 500 m de distância, seis picocélulas e dez femtocélulas, totalizando 37 células, além de largura de banda de 5 MHz e cinquenta blocos de recursos por célula. O cenário de estresse difere do de referência

exclusivamente pela densidade de usuários, mantendo inclusive o prazo de 60 slots. Essa restrição é deliberada e metodologicamente necessária: uma versão anterior do cenário alterava simultaneamente densidade, prazo e topologia, o que impedia atribuir o comportamento observado a qualquer fator isolado. Com a topologia e o prazo congelados, toda diferença em relação ao cenário de referência é atribuível à carga.

Duas consequências valem para todo o capítulo. A primeira é que os valores absolutos não são comparáveis entre cenários, uma vez que cada um representa um regime distinto de ocupação. A segunda, que é a que preserva as conclusões, é que as comparações entre escalonadores permanecem íntegras: dentro de cada cenário, os cinco enfrentam a mesma topologia, a mesma distribuição de usuários e a mesma rotina de extração de métricas, de modo que qualquer efeito de cobertura ou de saturação incide igualmente sobre todos e se cancela na comparação. As 36 comparações estatísticas da Seção 6.5 não são afetadas.

6.5 Análise Consolidada

As quatro tabelas anteriores somam 36 comparações estatísticas: quatro cenários, três linhas de base nativas e três métricas. O *Proportional Fair*, incluído nessas tabelas como referência adicional, é analisado separadamente na Seção 6.6, por constituir a base sobre a qual o DRLQoSScheduler é construído. A Tabela 6.12 consolida os vereditos.

Tabela 6.12 - Consolidação das 36 comparações estatísticas

Cenário	Vazão	Equidade	Cumprimento de prazo	Total a favor
Base	3 a favor	2 a favor, 1 empate	3 a favor	8 de 9
Carga	2 a favor, 1 empate	3 empates	3 a favor	5 de 9
Mobilidade	3 empates	2 a favor, 1 empate	3 a favor	5 de 9
Estresse	2 a favor, 1 empate	2 a favor, 1 empate	3 a favor	7 de 9
Total				25 de 36

Fonte: Elaborada pelo autor.

O resultado agregado é de 25 comparações a favor do DRLQoSScheduler, 0 contra e 11 empates estatísticos. A Figura 6.4 apresenta a mesma informação em forma de matriz, com os p-valores.

Figura 6.4 - Matriz de significância das 36 comparações

		Vazão	Equidade	Cumprimento de prazo
Referência	Round Robin	<0,0001 a favor	0,8346 empate	<0,0001 a favor
	Best CQI	<0,0001 a favor	<0,0001 a favor	<0,0001 a favor
	QoS Aware	<0,0001 a favor	<0,0001 a favor	<0,0001 a favor
Carga elevada	Round Robin	0,0070 a favor	0,7164 empate	0,0020 a favor
	Best CQI	0,3432 empate	0,6108 empate	<0,0001 a favor
	QoS Aware	0,0044 a favor	0,1419 empate	<0,0001 a favor
Mobilidade	Round Robin	0,8506 empate	0,1440 empate	0,0153 a favor
	Best CQI	0,5045 empate	0,0081 a favor	0,0252 a favor
	QoS Aware	0,8789 empate	<0,0001 a favor	<0,0001 a favor
Estresse	Round Robin	<0,0001 a favor	<0,0001 a favor	<0,0001 a favor
	Best CQI	0,6369 empate	<0,0001 a favor	<0,0001 a favor
	QoS Aware	0,0001 a favor	0,5554 empate	<0,0001 a favor

A favor do DRLQoSScheduler ($p < 0,05$)
 Empate estatístico ($p \geq 0,05$)

Fonte: Elaborada pelo autor.

Três leituras se destacam.

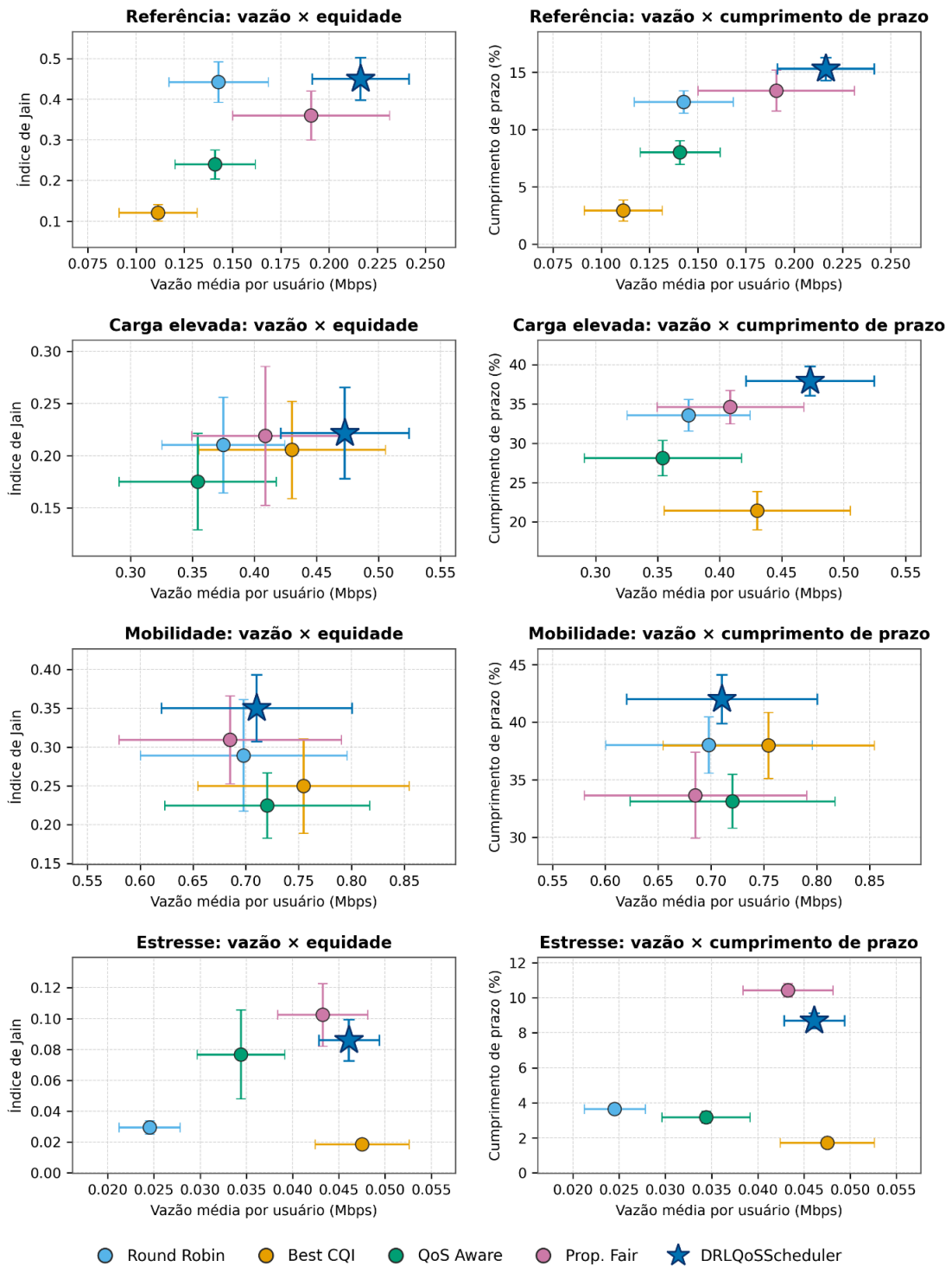
1) Ausência de derrota. Em nenhuma das 36 comparações uma linha de base supera o DRLQoSScheduler com significância estatística. Nos casos em que uma linha de base apresenta média melhor (a vazão do *Best CQI* no cenário de carga elevada e no de mobilidade, por exemplo), a diferença permanece dentro do intervalo de confiança. Essa é a afirmação mais forte que os dados sustentam, e é mais robusta que a contagem de vitórias: o DRLQoSScheduler nunca é pior nos testes, em nenhum eixo, em nenhuma condição avaliada.

2) O cumprimento de prazo é o eixo mais consistente. Das doze comparações dessa métrica, todas as doze favorecem o DRLQoSScheduler com significância. É o único eixo com desempenho uniforme nos quatro cenários, e sugere que a dimensão de urgência de latência da ação é a que o agente explora de forma mais efetiva.

3) A equidade é o eixo mais dependente do cenário. Das doze comparações, seis favorecem o DRLQoSScheduler e seis resultam em empate. O padrão tem estrutura: os empates

concentram-se nas comparações contra o *Round Robin*, que é a referência de equidade cega, e no cenário de carga elevada, em que a competição por recursos reduz a margem de redistribuição e os cinco escalonadores convergem para valores próximos. Contra o Best CQI a vantagem é significativa em três dos quatro cenários, e contra o QAS em dois.

Figura 6.5 - Planos de compromisso: vazão versus equidade e vazão versus cumprimento de prazo

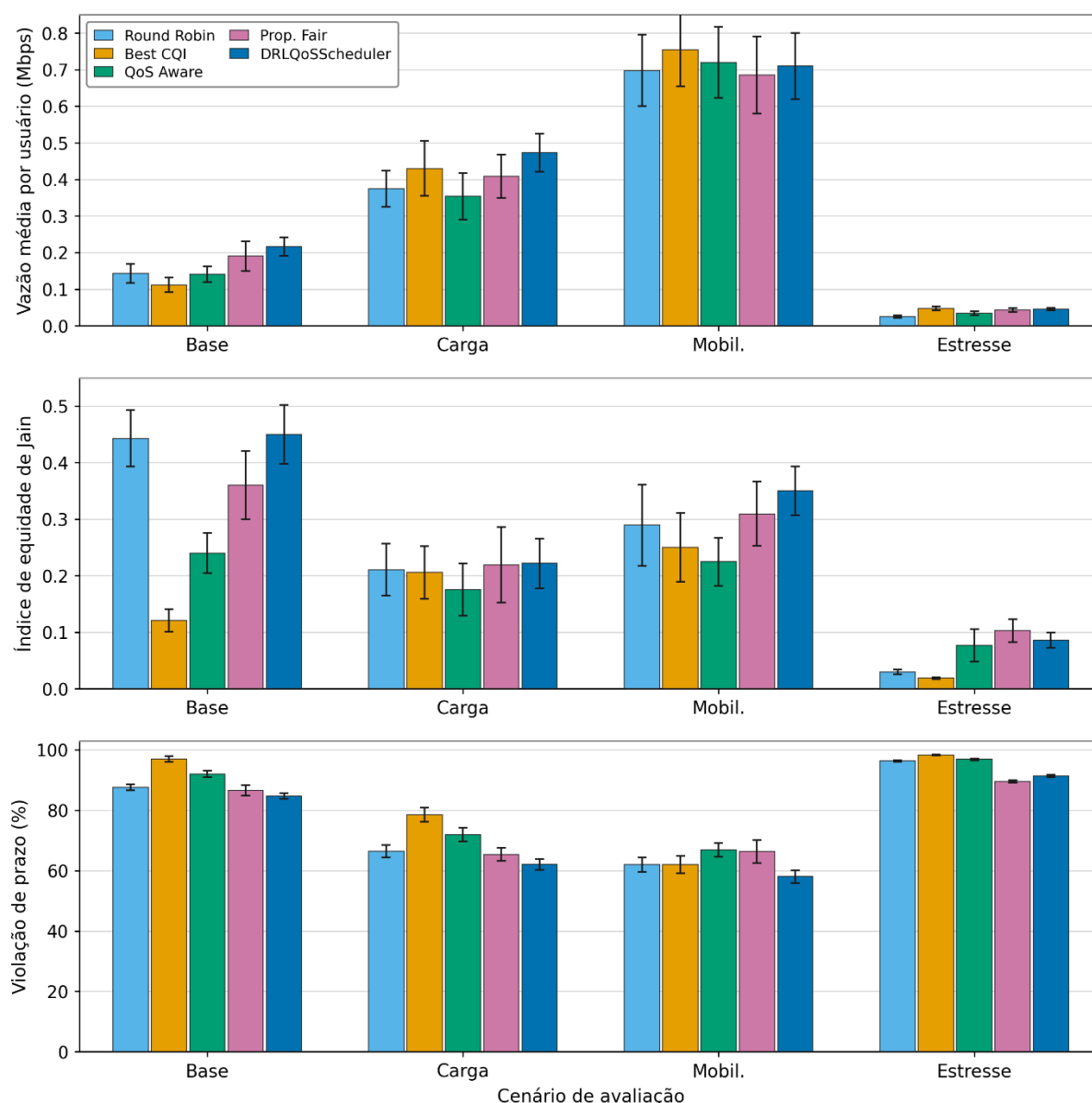


Fonte: Elaborada pelo autor.

A Figura 6.5 apresenta o plano de compromisso entre vazão e equidade, e entre vazão e cumprimento de prazo, com um marcador por escalonador em cada cenário. A representação torna visível o argumento central da proposta: o DRLQoSScheduler ocupa a região favorável dos dois planos em todos os cenários, ao passo que cada linha de base ocupa a região favorável de um eixo ao custo do outro.

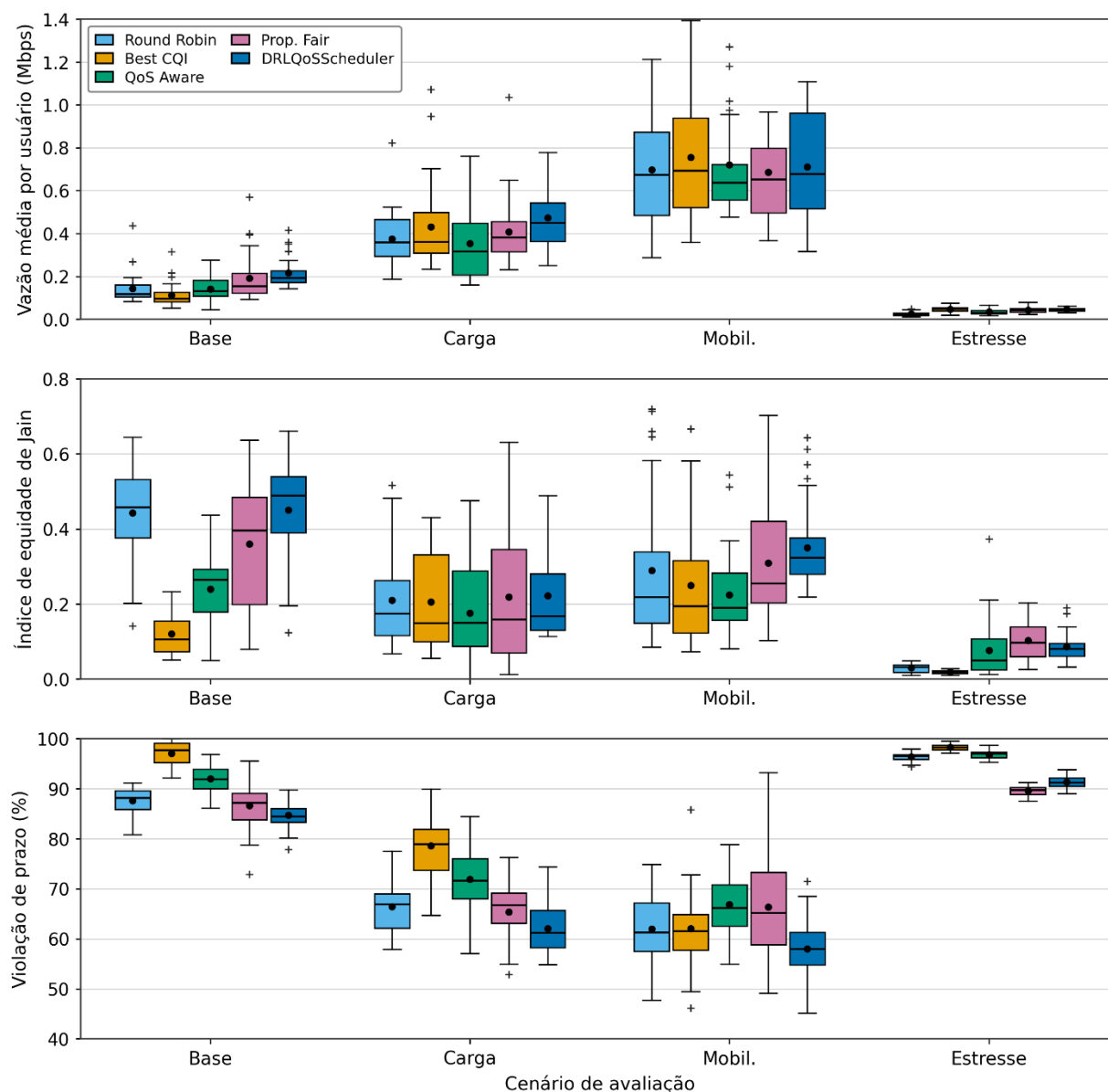
A Figura 6.6 apresenta a comparação por métrica em barras agrupadas, com os intervalos de confiança de 95%, e a Figura 6.7 mostra a dispersão entre as 30 rodadas, o que permite verificar que os empates reportados correspondem a distribuições efetivamente sobrepostas, e não a escassez de dados.

Figura 6.6 - Comparação por métrica e cenário, com intervalos de confiança de 95%



Fonte: Elaborada pelo autor.

Figura 6.7 - Dispersão das métricas entre as 30 rodadas independentes



Fonte: Elaborada pelo autor.

Nota: quatro observações do cenário de mobilidade, situadas entre 1,41 e 1,86 Mbps, excedem o limite superior do eixo do painel de vazão e não são exibidas.

6.6 Linha de Base Proportional Fair

As três linhas de base das seções anteriores são as disponíveis nativamente na plataforma de simulação. O *Proportional Fair*, referência mais usual na literatura de escalonamento e discutido na Seção 3.7.3, não integra o catálogo de escalonadores do simulador na versão utilizada.

Sua ausência seria uma lacuna relevante, porque a métrica de prioridade do DRLQoSScheduler combina a base proporcional-justa com os fatores de urgência herdados do QAS. As duas referências delimitam a proposta por lados opostos: o QAS representa os fatores de QoS com pesos fixos e sem memória proporcional-justa, e o Proportional Fair representa a base sem qualquer ponderação de QoS. Sem esta última, não é possível separar a contribuição da base proporcional-justa da contribuição da adaptação exercida pelo agente.

A linha de base foi obtida sem implementar um novo escalonador, executando o próprio DRLQoSScheduler com a ação fixada em $[0,01; 0; 0]$. Conforme demonstrado na Seção 5.4, com o peso de latência nulo o fator de urgência correspondente vale um para todos os usuários; o mesmo ocorre com o fator de confiabilidade quando seu peso é nulo; e uma meta de equidade de 0,01 encerra a correção iterativa na primeira verificação. A métrica da Equação 5.2 reduz-se então a $T(u,r)/T(\bar{u})$, que é exatamente a métrica do *Proportional Fair*. A comparação resultante é imune à objeção de que eventuais diferenças decorreriam de detalhes de implementação distintos, pois trata-se do mesmo código com o controle desativado.

A Tabela 6.13 reúne o desempenho do *Proportional Fair* nos quatro cenários, e a Tabela 6.14 consolida a comparação com o DRLQoSScheduler, acrescentando a diferença relativa em cada métrica. Os mesmos valores constam também das Tabelas 6.4 a 6.11, ao lado dos demais escalonadores.

Tabela 6.13 - Desempenho do *Proportional Fair* nos quatro cenários

Cenário	Vazão média por usuário (Mbps)	Índice de equidade de Jain	Violação de prazo (%)
Base	$0,1906 \pm 0,0406$	$0,3599 \pm 0,0602$	$86,59 \pm 1,78$
Carga	$0,4085 \pm 0,0592$	$0,2191 \pm 0,0667$	$65,39 \pm 2,13$
Mobilidade	$0,6851 \pm 0,1052$	$0,3095 \pm 0,0566$	$66,33 \pm 3,74$
Estresse	$0,0432 \pm 0,0049$	$0,1025 \pm 0,0203$	$89,56 \pm 0,38$

Fonte: Elaborada pelo autor.

O resultado é de 3 comparações a favor do DRLQoSScheduler, 1 contra e 8 empates, em 12 comparações. A única derrota registrada no capítulo ocorre nesta comparação, no cumprimento de prazo do cenário de estresse, e é analisada adiante. A proporção de vantagens é sensivelmente menor que a observada contra as três linhas de base nativas, que foi de 25 em 36.

Esta é a comparação mais exigente do capítulo, e a que melhor delimita a contribuição da proposta. As demais linhas de base diferem do DRLQoSScheduler em dois aspectos simultâneos: a métrica de prioridade e a adaptação dos parâmetros. O *Proportional Fair* difere

em apenas um. A diferença observada aqui é, portanto, atribuível exclusivamente ao controle adaptativo exercido pelo agente.

Tabela 6.14 - DRLQoS Scheduler comparado ao Proportional Fair

Cenário	Métrica	DRLQoS Scheduler	Prop. Fair	Diferença	p-valor	Veredito
Base	Vazão (Mbps)	0,2163	0,1906	13,5%	0,2754	empate
	Equidade de Jain	0,4501	0,3599	25,1%	0,0240	a favor
	Violação de prazo (%)	84,72	86,59	2,2%	0,0654	empate
Carga	Vazão (Mbps)	0,4728	0,4085	15,7%	0,0997	empate
	Equidade de Jain	0,2217	0,2191	1,2%	0,9471	empate
	Violação de prazo (%)	62,07	65,39	5,1%	0,0198	a favor
Mobilidade	Vazão (Mbps)	0,7102	0,6851	3,7%	0,7124	empate
	Equidade de Jain	0,3501	0,3095	13,1%	0,2479	empate
	Violação de prazo (%)	58,01	66,33	12,5%	0,0002	a favor
Estresse	Vazão (Mbps)	0,0461	0,0432	6,6%	0,3219	empate
	Equidade de Jain	0,0860	0,1025	16,1%	0,1705	empate
	Violação de prazo (%)	91,31	89,56	1,9%	< 0,0001	contra

Fonte: Elaborada pelo autor.

A leitura conjunta com a Seção 6.5 leva a uma conclusão que convém enunciar sem atenuação: parte substancial da vantagem do DRLQoS Scheduler sobre o *Round Robin*, o *Best CQI* e o *QAS* decorre da métrica proporcional-justa que lhe serve de base, e não do aprendizado. O *Proportional Fair*, sozinho, já supera o *Best CQI* e o *QAS* em várias comparações no cenário de referência. Atribuir todo o ganho ao componente de aprendizado seria incorreto.

O que o aprendizado acrescenta é mensurável e tem padrão definido. Das três vantagens significativas sobre o *Proportional Fair*, duas estão no cumprimento de prazo, nos cenários de carga elevada e de mobilidade, e a terceira está na equidade do cenário de referência. É coerente com o observado na Seção 6.5, em que o prazo é o eixo de vantagem mais consistente, e com o mecanismo discutido na Seção 6.3.3: o *Proportional Fair* é cego a prazo, ao passo que o DRLQoS Scheduler dispõe do parâmetro de urgência de latência e pode calibrá-lo conforme o estado da rede.

O cenário de estresse inverte esse padrão e produz a única derrota do capítulo. Em vazão e em equidade os dois escalonadores empatam, com p de 0,3219 e de 0,1705 respectivamente. Em cumprimento de prazo, porém, o *Proportional Fair* obtém 89,56% de violação contra 91,31% do DRLQoS Scheduler, diferença de 1,7 ponto percentual com p inferior a 0,0001. Sob

saturação, portanto, a capacidade de ajustar os parâmetros deixa de produzir vantagem sobre a métrica proporcional-justa pura e passa a produzir uma pequena desvantagem no eixo temporal. A leitura mais provável é que, com cerca de 8,6 usuários por célula, a margem de redistribuição se esgota: qualquer priorização de tráfego urgente implica retirar recursos de outro usuário igualmente carente, e o ajuste dos pesos deixa de ter efeito líquido positivo. O resultado delimita o alcance da proposta e é retomado no capítulo de conclusão.

Nos demais casos a diferença não atinge significância. No cenário de mobilidade, em particular, os dois escalonadores empatam em vazão e em equidade, e a única vantagem é o cumprimento de prazo.

Uma observação adicional decorre desta seção. O *Proportional Fair* apresenta, no cenário de referência, vazão de 0,1906 Mbps, superior tanto à do *Round Robin* (0,1427 Mbps) quanto à do *Best CQI* (0,1113 Mbps). Como o *Proportional Fair* é, por construção, um compromisso entre esses dois comportamentos, seu valor deveria situar-se entre eles. O resultado é mais um sintoma da anomalia do *Best CQI* discutida na Seção 6.3.1, e é coerente com o mecanismo descrito naquela seção.

Uma ressalva de escopo se aplica a esta linha de base. O limite inferior imposto à meta de equidade é 0,01, ao passo que o menor valor possível do índice de Jain é $1/N$. No cenário de estresse, com cerca de 320 usuários, esse mínimo é de aproximadamente 0,003, inferior ao limite. Em slots nos quais a alocação for suficientemente concentrada, a correção de equidade pode acionar e realizar algumas transferências, de modo que a linha de base nesse cenário específico é uma aproximação do *Proportional Fair* estrito, e não sua reprodução exata.

6.7 Custo Computacional

As medições de tempo reportadas nesta seção foram obtidas em estação de trabalho com processador Intel Core i5-1135G7 de quatro núcleos a 2,40 GHz, 8 GB de memória RAM a 3200 MT/s e unidade gráfica integrada Intel Iris Xe, sob Windows 11 e MATLAB R2025b. Não foi empregada unidade de processamento gráfico dedicada: tanto o treinamento quanto a inferência executaram exclusivamente em unidade central de processamento, com as cinco redes do ensemble avaliadas sequencialmente. Essa configuração é relevante para a leitura dos valores a seguir e é também o que fundamenta o orçamento computacional discutido na Seção 5.11, que motivou a redução do cenário de treinamento.

A viabilidade prática do DRLQoSScheduler depende do custo de inferência ser desprezível em relação ao intervalo de decisão, conforme o argumento da Seção 5.10.

A Tabela 6.15 apresenta a medição do tempo de inferência do *ensemble*. O protocolo descarta as cinquenta primeiras chamadas como aquecimento, eliminando o custo de compilação sob demanda do ambiente de execução, e remove as medições acima do percentil 99, 10 das 1000, para atenuar o efeito de interrupções do sistema operacional.

A inferência do *ensemble* consome 7,1429 ms, com intervalo de confiança de 0,1235 ms, dispersão de 1,7% em torno da média, o que indica medição estável. A proximidade entre média e mediana confirma que a distribuição não é dominada por valores extremos após o descarte.

Tabela 6.15 - Tempo de inferência do ensemble de cinco políticas

Grandeza	Valor
Medições realizadas	1000
Medições válidas após descarte do percentil 99	990
Tempo médio por inferência do ensemble	7,1429 ms $\pm$ 0,1235 ms
Mediana	7,0583 ms
Mínimo observado	5,1920 ms
Máximo observado	13,5429 ms
Tempo médio por rede individual	1,4286 ms
Intervalo entre decisões (20 slots de 1 ms)	20 ms
Fração do intervalo de decisão consumida	35,7%
Custo amortizado por slot	0,3571 ms

Fonte: Elaborada pelo autor.

A leitura correta desse número exige atenção ao intervalo de decisão, e não ao intervalo de *slot*. O escalonador executa a cada *slot* de 1 ms, mas o agente é consultado a cada 20 *slots*, conforme a Seção 5.3.4. O orçamento temporal disponível para a inferência é, portanto, de 20 ms, e não de 1 ms. Com 7,1429 ms medidos, a inferência consome 35,7% do intervalo entre decisões e conclui com margem antes do passo seguinte. Amortizado sobre os 20 *slots* do intervalo, o custo corresponde a 0,3571 ms por *slot*.

Esse resultado qualifica, e não confirma, a expectativa registrada na Seção 5.10. Aquela seção antecipava custo de ordem submilissegundo por decisão; o valor medido é uma ordem de grandeza superior. A viabilidade se sustenta pelo intervalo de decisão ampliado, o agente dispõe de 20 ms, e não de 1 ms, e não pela pequenez do custo absoluto. Caso o passo de decisão fosse reduzido a um único *slot*, como exigiria uma formulação que alocasse RBs diretamente, o custo excederia o intervalo disponível em mais de sete vezes. A decisão de projeto de amortizar a inferência ao longo de vinte *slots*, discutida na Seção 5.2.1, é portanto condição necessária da viabilidade, e não um refinamento.

Dois fatores atenuam o valor medido em uma implementação real. O primeiro é o ambiente de execução: a medição foi feita em MATLAB, com as cinco redes avaliadas sequencialmente a 1,4286 ms cada, sem paralelização e sem otimização de inferência. Uma implementação em código nativo, com precisão numérica reduzida e avaliação paralela das cinco redes, reduziria o custo substancialmente. O segundo é o próprio *ensemble*: uma política individual custaria um quinto do medido, e esse custo adicional é o preço, modesto e amortizado, da maior robustez que a seleção por ranqueamento da Seção 6.2 já demonstrou.

Cabe delimitar o alcance da medição. Ela demonstra que o componente de aprendizado é compatível com a escala temporal adotada neste trabalho, o que responde à objeção mais frequente ao emprego de aprendizado por reforço em decisões de camada de acesso. Não constitui avaliação de implementação embarcada: medições em processador de propósito geral não transferem para o hardware de uma estação-base, onde memória disponível, precisão numérica e integração com o processamento de banda base determinariam o custo efetivo.

Na campanha de avaliação descrita na Seção 5.9, em que a ação é determinada uma vez por rodada de 1 000 *slots*, o custo amortizado cai para 0,007143 ms por *slot*. Esse valor não deve ser tomado como o custo do método em operação, pois reflete o regime de adaptação reduzido da avaliação, e não o regime de projeto.

A medição anterior cobre apenas o agente. O escalonador, porém, executa a cada *slot* de 1 ms, e não a cada 20 *slots*, de modo que dispõe de um orçamento vinte vezes menor. A Tabela 6.16 apresenta a medição do tempo de execução do método de escalonamento, por *slot* e por célula, obtida por instrumentação direta do simulador nos quatro cenários.

Tabela 6.16 - Tempo de execução do escalonador por slot e por célula

Cenário	Usuários ativos por célula	Tempo médio (ms)	Percentil 95 (ms)	Máximo (ms)	Slots acima de 1 ms
Referência	4,4	2,2230	4,3404	67,7692	91,72%
Carga elevada	2,4	1,2803	2,7004	26,1451	51,24%
Mobilidade	1,8	0,9069	1,8705	6,7398	27,89%
Estresse	22,8	9,0902	18,2533	82,3632	85,39%

Fonte: Elaborada pelo autor.

O tempo médio situa-se entre 0,9069 ms e 9,0902 ms. Apenas o cenário de mobilidade permanece abaixo do intervalo de 1 ms do slot; nos demais o valor o excede. Antes de interpretar esse resultado como inviabilidade, convém examinar como ele se comporta com a

carga. O custo cresce linearmente com o número de usuários ativos. O ajuste por mínimos quadrados dos quatro cenários resulta em:

$$t = 0,3631 + 0,3837 \cdot N_{UE} [ms], \text{ com } R^2 = 0,9988, \quad (6.1)$$

o que confirma empiricamente a análise de complexidade $O(N_{RB} \cdot N_{UE})$ apresentada na Seção 5.4.6. Até aqui essa complexidade era apenas uma estimativa analítica; a medição em quatro cenários com densidades distintas a sustenta com coeficiente de determinação de 0,9988.

A decomposição do modelo separa duas parcelas. A parcela fixa, de 0,3631 ms, independe do número de usuários e corresponde a 10,8% do custo médio dos quatro cenários: é o percurso dos 50 blocos de recursos da grade, que ocorre em todo *slot*, somado à reinicialização da grade. A parcela variável, de 0,3837 ms por usuário ativo, corresponde a 7,67 μ s por par de usuário e bloco de recursos.

O valor absoluto é dominado por sobrecarga do ambiente de execução, e não por trabalho algorítmico. O laço interno que estima a vazão realiza duas chamadas de método por par de usuário e bloco de recursos, uma para o número de símbolos de dados e outra para a eficiência espectral associada ao indicador de qualidade de canal. Dividindo a parcela variável por essas duas chamadas, obtém-se 3,84 μ s por chamada, valor compatível com o custo típico de uma chamada de método em MATLAB (*Mathworks*) interpretado, entre 1 e 10 μ s. Em outras palavras, o tempo medido reflete predominantemente o custo de despacho do interpretador, e não a aritmética do escalonamento, que envolve cerca de 250 operações elementares por *slot*.

A correção iterativa de equidade não é o gargalo. A instrumentação separou essa etapa das demais e ela responde por 0,4% do tempo total no cenário de estresse e por valores entre 1,6% e 2,1% nos demais. A fração é menor justamente onde a carga é maior, porque o laço principal de alocação cresce com o número de usuários enquanto a correção encerra antecipadamente. A hipótese inicial, motivada pelo pior caso $O(N_{RB}^2 \cdot N_{UE})$ discutido na Seção 5.4.4, não se confirmou: as condições de parada antecipada encerram o laço nas primeiras iterações na quase totalidade dos *slots*.

Os valores máximos, entre 6,7 ms e 82,4 ms, são ordens de grandeza acima da mediana e não representam comportamento do algoritmo. Decorrem de interrupções do sistema operacional e da compilação sob demanda nas primeiras chamadas, artefatos usuais em medição de tempo em ambiente interpretado e não isolado.

A medição confirma a complexidade declarada e localiza o custo: linear na carga, com parcela fixa reduzida, de cerca de um décimo do tempo médio, e o restante dominado por

despacho de método. Ela não permite afirmar que o escalonador cabe no *slot* de 1 ms em uma estação-base real, nem o contrário: o MATLAB interpretado é instrumento inadequado para essa pergunta. A verificação exigiria implementação em código nativo e medição em hardware representativo, o que está fora do escopo deste trabalho e integra a agenda do Capítulo 7. Registre-se ainda que o tempo de execução não influencia nenhum resultado das seções anteriores: o simulador computa o efeito das decisões de escalonamento independentemente do tempo que o código leva para produzi-las.

Uma observação sobre a contagem de usuários. O número de usuários ativos por célula varia entre 1,8 e 22,8 nos quatro cenários. A relação com a densidade total não é proporcional: os usuários distribuem-se de forma desigual entre as 37 células da topologia, com concentração nos setores macrocelulares, e apenas aqueles que têm dados pendentes em buffer entram na alocação. Slots sem usuários ativos retornam antecipadamente e foram excluídos da estatística, o que torna a medida conservadora.

6.8 Discussão

Os resultados sustentam a hipótese central da dissertação, com qualificações que convém explicitar. O objetivo desta dissertação não foi maximizar uma métrica isolada, mas obter um escalonador equilibrado entre justiça e qualidade de serviço, capaz de conciliar equidade na distribuição de recursos com o atendimento às demandas de vazão e de prazo sob condições dinâmicas de rede. É esse equilíbrio, e não a supremacia em um único eixo, que os resultados sustentam. Em todos os quatro cenários, o DRLQoS Scheduler ocupa simultaneamente a região favorável dos planos de compromisso entre vazão e equidade e entre vazão e prazo (Figura 6.5), posição que nenhuma linha de base alcança. O *Round Robin* garante equidade ao custo do prazo, e o *QAS*, apesar de projetado para respeitar prazos, figura entre os dois piores em violação de prazo nos quatro cenários e não lidera nenhum dos dois eixos. O *Best CQI* troca equidade por vazão nos cenários de maior densidade, e no cenário de referência não obtém nem uma nem outra, pela razão discutida na Seção 6.3.1. O DRLQoS Scheduler é o único que não sacrifica um eixo pelo outro, e é essa propriedade que a ausência de qualquer derrota estatística nas 36 comparações traduz.

Esse equilíbrio deve ser lido em termos relativos, e não como atendimento absoluto a requisitos de serviço. Em todos os cenários as taxas de violação de prazo permanecem elevadas para todos os escalonadores, e nenhum deles as reduz a patamares compatíveis com serviços críticos. As causas, porém, diferem conforme o regime de ocupação, conforme discutido na

Seção 6.4.4. Nos cenários de maior densidade a limitação é de recurso: com poucos megahertz de banda divididos entre dezenas de usuários, não há grau de liberdade que faça toda a demanda caber dentro do prazo. No cenário de referência, ao contrário, a rede opera com cerca de um usuário por célula, e o piso da violação decorre principalmente dos usuários que nunca recebem alocação por estarem em posições de cobertura desfavorável. Em nenhum dos dois casos a política de escalonamento é o fator dominante. O que o DRLQoSScheduler oferece, e é o que se pode legitimamente reivindicar, é o melhor compromisso entre justiça e QoS disponível sob esses limites: a maior redução relativa de violação de prazo mantendo a maior equidade entre as referências nativas, de forma consistente e transferível a cenários não vistos.

O que os dados sustentam: O controle paramétrico adaptativo produz um escalonador que não é dominado por nenhuma das três linhas de base nativas em nenhuma condição avaliada. Em carga moderada, obtém vazão substancialmente superior mantendo equidade equivalente à do *Round Robin*, combinação que nenhuma linha de base alcança. Em carga elevada e sob saturação, converte sua liberdade de controle em cumprimento de prazo. Sob mobilidade, mantém a vantagem em equidade e prazo mesmo com a variância elevada que a mobilidade impõe. E faz isso em três cenários que não participaram do treinamento, sem qualquer reajuste, o que é a evidência mais relevante em favor da transferibilidade argumentada na Seção 5.5.4. O limite dessa afirmação está na comparação com o *Proportional Fair*, em que o cenário de saturação produz a única derrota do capítulo.

Delimitação do alcance dos resultados: os ganhos observados devem ser lidos dentro de fronteiras bem definidas. No cumprimento de prazo, o que se demonstra é a maior redução relativa de violações, não o atendimento a requisitos de uRLLC, inatingíveis por qualquer escalonador sob as condições simuladas, em que os fatores dominantes são a ocupação da topologia e a disponibilidade de recurso, e não a política de alocação (Seção 6.4.4). Em equidade, o DRLQoSScheduler iguala o *Round Robin*, referência de equidade cega, nos três cenários de ocupação moderada, e o supera com significância sob saturação, sempre entregando vazão substancialmente maior, combinação que nenhuma linha de base alcança. Em vazão, a paridade com o *Best CQI* sob carga elevada, mobilidade e saturação preserva a ausência de derrota que caracteriza as 36 comparações. Resta como fronteira genuína a dimensão de confiabilidade, implementada e validada nos limites, porém ainda não exercitada (Seção 5.4.3): sua ativação plena é a extensão mais imediata deste trabalho.

O comportamento anômalo do *QAS* é o achado colateral mais interessante. Nos quatro cenários, o *QAS* apresenta taxa de violação de prazo superior à do *Round Robin*, que ignora prazos por completo. A explicação proposta na Seção 6.3.3, a despriorização proporcional à

folga, que sob congestionamento posterga o tráfego crítico até um ponto sem recuperação, tem implicação prática além deste trabalho: sugere que mecanismos de urgência baseados em folga são contraproducentes em redes saturadas, e que a intensidade desse mecanismo deveria ser função da carga. É precisamente o que o DRLQoSScheduler faz ao tratar o peso de urgência como variável de controle, e ajuda a explicar por que o cumprimento de prazo é seu eixo de vantagem mais consistente.

Sobre a magnitude dos valores absolutos. A vazão média por usuário situa-se entre 0,02 e 0,75 Mbps conforme o cenário. Os valores decorrem da divisão de 5 MHz de banda entre dezenas de usuários e não devem ser lidos como desempenho de uma rede real; a comparação relevante é sempre entre escalonadores sob condições idênticas. Analogamente, os índices de Jain entre 0,02 e 0,45 refletem alocação concentrada, própria de rede congestionada.

6.9 Considerações Finais

Este capítulo avaliou o DRLQoSScheduler em quatro cenários de rede heterogênea, contra três escalonadores de referência, em 30 rodadas independentes por combinação, totalizando 36 comparações estatísticas. O resultado agregado é de 25 comparações a favor do DRLQoSScheduler, 0 contra e 11 empates.

O cenário de referência produziu o resultado mais expressivo: vazão superior às três linhas de base com significância, mantendo equidade estatisticamente equivalente à do *Round Robin*. Nos três cenários não vistos durante o treinamento, a vantagem se mantém, concentrando-se no cumprimento de prazo, o único eixo em que o DRLQoSScheduler supera todas as referências em todos os cenários.

A ausência de qualquer derrota estatística, em 36 comparações, é o resultado que melhor caracteriza a proposta: não se trata de um escalonador que maximiza uma métrica ao custo das demais, mas de um que evita ser ruim em qualquer uma delas, o que é precisamente o objetivo do controle adaptativo do ponto de operação.

7 CONCLUSÃO

Esta dissertação investigou o emprego de aprendizado por reforço profundo no escalonamento de recursos de *downlink* em redes 5G a partir de uma formulação específica: em vez de substituir a heurística de alocação por uma política aprendida, preservá-la e confiar ao agente apenas o ajuste contínuo, em tempo de execução, dos parâmetros que definem seu ponto de operação no espaço de compromissos entre vazão, equidade e cumprimento de prazo. O resultado dessa formulação é o DRLQoSScheduler, apresentado no Capítulo 5 e avaliado no Capítulo 6.

A questão que o trabalho se propôs a responder não foi qual escalonador maximiza uma métrica isolada, mas se o controle paramétrico adaptativo produz um escalonador equilibrado, capaz de conciliar equidade na distribuição de recursos com o atendimento a requisitos de vazão e de prazo sob condições dinâmicas de rede. Este capítulo retoma essa questão, sintetiza as evidências reunidas, delimita as contribuições e projeta as direções de continuidade.

7.1 Síntese dos Resultados

A campanha experimental compreendeu quatro cenários de rede heterogênea, três escalonadores de referência disponíveis na plataforma de simulação, uma linha de base *Proportional Fair* obtida por desativação do controle, e trinta rodadas independentes por combinação de cenário e escalonador. Disso resultaram 36 comparações estatísticas com as referências nativas e outras 12 em relação ao *Proportional Fair*, todas com intervalos de confiança de 95% e testes de significância bilaterais.

O resultado agregado com relação às três referências nativas é de 25 comparações a favor do DRLQoSScheduler, nenhuma contra e 11 empates estatísticos. A ausência de derrota é a afirmação mais forte que os dados sustentam, e é mais informativa que a contagem de vitórias: em nenhum eixo de desempenho, em nenhuma das condições avaliadas, uma linha de base nativa supera o DRLQoSScheduler com significância. Nos planos de compromisso apresentados na Seção 6.5, o DRLQoSScheduler ocupa simultaneamente a região favorável dos dois eixos em todos os cenários, posição que nenhuma referência alcança: o Best CQI prioriza vazão ao custo da equidade, o *Round Robin* garante equidade ao custo do prazo, e o QAS, apesar de projetado para respeitar prazos, não supera o *Round Robin* nessa métrica em nenhum dos quatro cenários. A ordenação no cenário de saturação sintetiza esse comportamento: ali o

DRLQoSScheduler ocupa a segunda posição nas três métricas simultaneamente, e é o único dos cinco escalonadores a consegui-lo.

O atendimento aos prazos estabelecidos atua como o eixo de maior vantagem operacional. As doze comparações dessa métrica, nos quatro cenários e contra as três referências nativas, favorecem o DRLQoSScheduler com significância estatística. A equidade é o eixo mais dependente do cenário, com seis vantagens e seis empates, estes concentrados nas comparações contra o *Round Robin*, que é a referência de equidade cega, e no cenário de carga elevada, em que os cinco escalonadores convergem para valores próximos e a margem de redistribuição se estreita. Três dos quatro cenários não participaram do treinamento, e a vantagem se mantém neles sem qualquer reajuste da política, o que constitui a evidência mais relevante em favor da transferibilidade argumentada na Seção 5.5.4.

A comparação com o *Proportional Fair* delimita a contribuição do componente de aprendizado com precisão maior, e é por isso a mais exigente do trabalho. As referências nativas diferem do DRLQoSScheduler em dois aspectos simultâneos, a métrica de prioridade e a adaptação dos parâmetros; o *Proportional Fair* difere em apenas um. O resultado nessa comparação é de 3 vantagens, 1 derrota e 8 empates. A conclusão que dela decorre foi enunciada sem atenuação na Seção 6.6 e convém repeti-la aqui: parte substancial da vantagem sobre as referências nativas decorre da métrica proporcional-justa que serve de base ao escalonador, e não do aprendizado. O que o aprendizado acrescenta é mensurável e tem padrão definido: duas das três vantagens estão no cumprimento de prazo, nos cenários de carga elevada e de mobilidade, eixo a que o *Proportional Fair* é cego, e a terceira está na equidade do cenário de referência. A única derrota registrada no trabalho ocorre nesta comparação, no cumprimento de prazo sob saturação, e delimita o alcance da proposta: com cerca de 8,6 usuários por célula a margem de redistribuição se esgota, e o ajuste dos parâmetros deixa de produzir efeito líquido positivo sobre o eixo temporal.

A medição de custo computacional respondeu à objeção mais frequente ao emprego de aprendizado por reforço em decisões de camada de acesso, e qualificou a expectativa registrada na Seção 5.10. A inferência do *ensemble* de cinco políticas consome 7,1429 ms, uma ordem de grandeza acima do antecipado, mas equivalente a 35,7% do intervalo de 20 ms entre decisões e a 0,3571 ms amortizados por slot. A viabilidade se sustenta, portanto, pelo intervalo de decisão ampliado, e não pela pequenez do custo absoluto: a decisão de projeto de amortizar a inferência ao longo de vinte slots é condição necessária do resultado, e não um refinamento. O escalonador propriamente dito, medido por instrumentação direta do simulador, apresenta custo linear no número de usuários ativos, com coeficiente de determinação de 0,9988, o que

confirma empiricamente a análise de complexidade da Seção 5.4.6. A correção iterativa de equidade, cujo pior caso motivara preocupação, responde por 0,4% do tempo total no cenário de saturação e por valores entre 1,6% e 2,1% nos demais.

7.2 Contribuições

O cerne da contribuição deste estudo consiste em sua formulação. A arquitetura proposta delega ao agente a função de controle de alto nível sobre uma heurística de domínio, abstendo-se da alocação direta de recursos. Tal modelagem não apenas preserva a complexidade e o tempo de execução do escalonador, como também restringe o espaço de ação a três dimensões contínuas, independentes do número de usuários e blocos, garantindo a viabilidade do treinamento. Esta abordagem preenche uma lacuna metodológica delineada no Capítulo 2, dado que a literatura aplica os agentes majoritariamente na alocação direta de recursos, negligenciando a parametrização adaptativa de heurísticas orientadas a QoS.

A segunda contribuição é metodológica e diz respeito ao isolamento da linha de base. O *Proportional Fair*, referência mais usual da literatura de escalonamento, não integra o catálogo da plataforma na versão utilizada. Em vez de implementar um escalonador adicional, com o risco de que diferenças de implementação contaminassem a comparação, a linha de base foi obtida executando o próprio DRLQoSScheduler com o controle desativado, condição sob a qual sua métrica de prioridade se reduz analiticamente à do PF. A comparação resultante é imune à objeção de que eventuais diferenças decorreriam de detalhes de implementação distintos, pois trata-se do mesmo código com um único componente desligado. O procedimento é transferível a qualquer trabalho que proponha controle adaptativo sobre uma heurística conhecida.

A terceira contribuição é o protocolo experimental. Dez agentes com sementes documentadas em vez de uma execução única, ranqueamento em avaliação sem ruído de exploração, *ensemble* de cinco políticas com pesos uniformes, trinta rodadas independentes por combinação e reporte sistemático de intervalos de confiança e de p-valores. O Capítulo 2 registrou a fragilidade de reprodutibilidade como lacuna recorrente na literatura de aprendizado por reforço aplicado a gestão de recursos de rádio; o protocolo adotado é uma resposta direta a ela, e permitiu que o trabalho reportasse, com igual rigor, tanto as vantagens quanto os empates e as limitações.

A quarta contribuição é a caracterização empírica do custo computacional, separando o agente do escalonador e reconhecendo que os dois operam sob orçamentos temporais distintos,

de 20 ms e de 1 ms respectivamente. A medição confirmou a complexidade declarada analiticamente, localizou a origem do custo e refutou a hipótese inicial de que a correção de equidade seria o gargalo. Trabalhos que empregam aprendizado por reforço em camada de acesso raramente reportam esse custo, e a sua omissão deixa em aberto a objeção mais imediata à aplicabilidade das propostas.

7.3 Trabalhos Futuros

As direções de continuidade decorrem, em boa parte, das delimitações declaradas na Seção 5.11 e dos pontos em aberto identificados no Capítulo 6. Reúnem-se aqui as que se consideram mais promissoras.

A quinta componente do vetor de estado emprega uma vazão de referência produzida pelo simulador que não é diretamente mensurável em uma rede em operação. A transposição do método para um sistema real exige substituí-la por uma estimativa construída a partir de informação efetivamente disponível na estação-base, como os relatórios de qualidade de canal e o histórico de alocação. Investigar quanto de desempenho se perde nessa substituição é condição para qualquer afirmação sobre aplicabilidade prática.

O orçamento de treinamento condicionou várias escolhas e é o candidato natural a revisão. Treinar os dez agentes no cenário completo, em vez do reduzido, exigiria cerca de 7 800 horas de processamento, contra as 420 efetivamente empregadas. Além do cenário, permanecem em aberto um estudo de suficiência do número de episódios e uma busca sistemática de hiperparâmetros, ambos dispensados pelo mesmo motivo. A ausência de convergência observada na Seção 6.2.1 confere urgência a essa agenda: é preciso determinar se a estagnação da recompensa decorre de orçamento insuficiente, da formulação da função de recompensa, da escala temporal do passo de decisão ou da inadequação do próprio DDPG ao problema, e a distinção entre essas hipóteses só se obtém por experimentação controlada.

No plano do modelo de rádio, duas extensões se destacam. A configuração de antena adotada é de fluxo único por setor; a passagem a múltiplas camadas espaciais alteraria a estrutura da estimação de vazão por RB e, por consequência, a métrica de prioridade sobre a qual o controle atua. E o escalonamento opera por célula, sem qualquer coordenação com as vizinhas, de modo que a interferência entre células discutida na Seção 3.5.3 é tratada como perturbação exógena, e não como variável sujeita a decisão.

A direção de mais longo alcance é a de distribuir a inteligência pela rede. Neste trabalho o controle é exercido por um agente único, consultado externamente, cuja política é aplicada de

maneira uniforme às células. A evolução natural é embarcar um agente em cada estação-base, de modo que cada nó decida localmente seu próprio ponto de operação a partir do estado que observa. Duas questões novas surgem dessa mudança. A primeira é de coordenação: decisões tomadas de forma independente em células vizinhas interagem pela interferência mútua, o que converte o problema em um jogo de múltiplos agentes, com a não-estacionariedade e a atribuição de crédito que lhe são próprias, e exige mecanismos de troca de informação entre nós ou o paradigma de treinamento centralizado com execução distribuída; os trabalhos multiagente examinados na Seção 2.4 oferecem ponto de partida, ainda que nenhum deles trate de controle paramétrico de heurística. A segunda é de implantação: um agente embarcado precisa caber no orçamento de memória e de tempo do hardware da estação-base e ser atualizável sem interrupção do serviço, requisitos que a arquitetura de controlador inteligente de rede de acesso, prevista nas propostas de rede de acesso aberta, endereça de forma explícita. Essa direção antecipa a visão de rede nativamente inteligente associada à sexta geração, na qual a inteligência não é um módulo acoplado ao escalonador de uma célula, mas atributo de cada estação-base, e a coordenação emerge da interação entre nós que aprendem. O presente trabalho contribui a esse programa com uma peça delimitada: uma política de controle paramétrico que já demonstrou transferir-se para condições não vistas sem reajuste, propriedade necessária a qualquer agente destinado a operar em nós heterogêneos e sob cargas que não conheceu durante o treinamento.

7.4 Considerações Finais

A alocação de recursos em redes celulares caracteriza-se como um problema de otimização multiobjetivo conflituoso. O balanço entre as métricas de vazão, equidade e garantia de latência exige um ajuste dinâmico, uma vez que o ponto de operação ideal depende de condições estocásticas de carga, mobilidade e composição do tráfego. Heurísticas clássicas abordam esse compromisso utilizando parâmetros rigidamente fixados em tempo de projeto, o que lhes confere ótimo desempenho sob as condições nominais de calibração, porém severa degradação em cenários não previstos. É precisamente essa inflexibilidade paramétrica, e não a fundamentação da heurística em si, que o presente trabalho se propõe a solucionar.

As evidências atestam o êxito da formulação proposta, uma vez que o escalonador desenvolvido se mantém não dominado por nenhuma das referências nativas em todos os cenários avaliados, ocupando consistentemente a região favorável de compromisso e transferindo essa resiliência para condições inéditas sem a necessidade de reajustes. A ausência

de inferioridade estatística nas 36 comparações contra as referências nativas comprova que o modelo não exaure uma métrica em favor de outra, cumprindo o objetivo de controle adaptativo. Entretanto, o rigor metodológico exige a delimitação precisa desse alcance: constatou-se que parcela expressiva dessa vantagem deriva da métrica proporcional-justa adjacente, e que o ganho estritamente atribuível ao aprendizado concentra-se na garantia das restrições de latência sob carga moderada, dissipando-se sob saturação, condição em que a proposta empata com o *Proportional Fair* em vazão e em equidade e apresenta pequena desvantagem no cumprimento de prazo. Adicionalmente, reportam-se limitações pertinentes, como a ausência de convergência plena no limite computacional estipulado, o que resultou na inércia de uma dimensão de ação, e a falta de aferição empírica direta da adaptação contínua nesta bateria de testes. Longe de serem meras ressalvas formais, tais apontamentos configuram questões científicas legítimas, estruturadas como direcionamentos futuros na agenda da Seção 7.3.

Em síntese, a presente dissertação demonstra que a modelagem dos parâmetros de uma heurística de escalonamento como variáveis de controle submetidas a aprendizado constitui uma estratégia exequível, mensurável e estritamente compatível com as restrições de tempo de execução típicas de uma rede de acesso. Tal abordagem preserva as propriedades analíticas fundamentais das heurísticas, notadamente a previsibilidade e a eficiência computacional, enquanto mitiga sua principal deficiência: a incapacidade de adaptação a cenários não previstos em projeto. A contribuição consolida um avanço objetivo rumo ao paradigma de redes autônomas, tendo como desdobramento natural a descentralização dessa inteligência computacional pelas estações-base, o que delineia a principal diretriz de continuidade desta pesquisa.

REFERÊNCIAS

3GPP. TR 36.814: Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects. Version 9.0.0, Release 9. Sophia Antipolis: 3rd Generation Partnership Project, 2010.

3GPP. TR 38.901: Study on channel model for frequencies from 0.5 to 100 GHz. Version 16.1.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2020.

3GPP. TS 23.501: System architecture for the 5G System (5GS). Version 16.6.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2020.

3GPP. TS 36.213: Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures. Version 16.8.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2022.

3GPP. TS 38.211: NR; Physical channels and modulation. Version 16.2.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2020.

3GPP. TS 38.213: NR; Physical layer procedures for control. Version 16.2.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2020.

3GPP. TS 38.214: NR; Physical layer procedures for data. Version 16.9.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2022.

3GPP. TS 38.300: NR; NR and NG-RAN Overall description; Stage 2. Version 16.9.0, Release 16. Sophia Antipolis: 3rd Generation Partnership Project, 2022.

AGARWAL, B.; TOGOU, M. A.; RUFFINI, M.; MUNTEAN, G.-M. A comprehensive survey on radio resource management in 5G HetNets: current solutions, future trends and open issues. *IEEE Communications Surveys & Tutorials*, v. 24, n. 4, p. 2495-2534, 2022. DOI: 10.1109/COMST.2022.3207967.

ALQWIDER, W.; RAHMAN, T. F.; MAROJEVIC, V. Deep Q-Network for 5G NR downlink scheduling. In: IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS WORKSHOPS (ICC WORKSHOPS), 2022, Seul. *Proceedings [...]*. Seul: IEEE, 2022. DOI: 10.1109/ICCWorkshops53468.2022.9814547.

ALWARAFY, A.; ABDALLAH, M.; ÇİFTLER, B. S.; AL-FUQAHA, A.; HAMDI, M. The frontiers of deep reinforcement learning for resource management in future wireless HetNets: techniques, challenges, and research directions. *IEEE Open Journal of the Communications Society*, v. 3, p. 322-365, 2022. DOI: 10.1109/OJCOMS.2022.3153226.

ALZUBAIDI, O. T. H.; ALHEEJAWI, S.; HINDIA, M. N.; DIMYATI, K. Interference mitigation strategies in beyond 5G wireless systems: a review. *Electronics*, v. 14, n. 11, 2237, 2025. DOI: 10.3390/electronics14112237.

ANAND, A.; DE VECIANA, G.; SHAKKOTTAI, S. Joint scheduling of URLLC and eMBB traffic in 5G wireless networks. *IEEE/ACM Transactions on Networking*, v. 28, n. 2, p. 477-490, abr. 2020. DOI: 10.1109/TNET.2020.2968373.

ANAND, D.; MUNTEAN, G.-M. Mitigating interferences in 5G O-RAN HetNets through ML-driven xAPP to enhance users' QoS. *IEEE Transactions on Network and Service Management*, v. 23, p. 2866-2883, 2026. DOI: 10.1109/TNSM.2026.3667462.

BUDHIRAJA, I.; KUMAR, N.; TYAGI, S. Deep-reinforcement-learning-based proportional fair scheduling control scheme for underlay D2D communication. *IEEE Internet of Things Journal*, v. 8, n. 5, p. 3143-3156, mar. 2021. DOI: 10.1109/JIOT.2020.3014926.

CHANG, H.-H.; SREE, R. B. S.; CHEN, H.; ZHANG, J.; LIU, L. MADRL based scheduling for 5G and beyond. Blacksburg: Virginia Tech; Dallas: Samsung Research America, 2022.

DOGRA, A.; JHA, R. K.; JAIN, S. A survey on beyond 5G network with the advent of 6G: architecture and emerging technologies. *IEEE Access*, v. 9, p. 67512-67547, 2021. DOI: 10.1109/ACCESS.2020.3031234.

DOMINGOS, G. C.; SILVA, É. R. da. Controle indireto por ensemble DDPG para escalonamento multiobjetivo em HetNets 5G com tráfego misto eMBB/uRLLC. In: SIMPÓSIO

BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS, 44., 2026, Salvador. Anais [...]. Salvador: SBrT, 2026. No prelo.

ERUNKULU, O. O.; ZUNGERU, A. M.; LEBEKWE, C. K.; MOSALAOSI, M.; CHUMA, J. M. 5G mobile communication applications: a survey and comparison of use cases. *IEEE Access*, v. 9, p. 97251-97295, 2021. DOI: 10.1109/ACCESS.2021.3093213.

GARBAZZA, I. E. *Uso de Deep Q-Learning para alocação de recursos uplink em redes heterogêneas LTE-A/5G*. 2025. Tese (Doutorado em Engenharia Elétrica) - Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia, Uberlândia, 2025. DOI: 10.14393/ufu.te.2025.374.

GULIA, S.; AHMAD, A.; SINGH, S.; GUPTA, M. D. Interference management in backhaul constrained 5G HetNets through coordinated multipoint. *Computers and Electrical Engineering*, v. 100, 107982, 2022. DOI: 10.1016/j.compeleceng.2022.107982.

HAO, Y.; LI, F.; ZHAO, C.; YANG, S. Delay-oriented scheduling in 5G downlink wireless networks based on reinforcement learning with partial observations. *IEEE/ACM Transactions on Networking*, v. 31, n. 1, p. 380-394, fev. 2023. DOI: 10.1109/TNET.2022.3194953.

HU, H.; WU, D.; ZHOU, F.; ZHU, X.; HU, R. Q.; ZHU, H. Intelligent resource allocation for edge-cloud collaborative networks: a hybrid DDPG-D3QN approach. *IEEE Transactions on Vehicular Technology*, v. 72, n. 8, p. 10696-10709, ago. 2023. DOI: 10.1109/TVT.2023.3253905.

HUANG, Y.; HOU, Y. T.; LOU, W. DELUXE: a DL-based link adaptation for URLLC/eMBB multiplexing in 5G NR. *IEEE Journal on Selected Areas in Communications*, v. 40, n. 1, p. 143-162, jan. 2022. DOI: 10.1109/JSAC.2021.3126084.

ITU-R. Report ITU-R M.2410-0: minimum requirements related to technical performance for IMT-2020 radio interface(s). Genebra: União Internacional de Telecomunicações, 2017.

JAIN, R. *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. New York: John Wiley & Sons, 1991.

KAEHLING, L. P.; LITTMAN, M. L.; MOORE, A. W. Reinforcement learning: a survey. *Journal of Artificial Intelligence Research*, v. 4, p. 237-285, 1996. DOI: 10.1613/jair.301.

KUMBI, G. G.; SHAHID, S. M.; DUONG, T. M.; KWON, S. Mobility and QoS-aware energy efficient scheduling for highway environments using deep reinforcement learning. *IEEE Transactions on Wireless Communications*, v. 25, p. 10921-10934, 2026. DOI: 10.1109/TWC.2025.3650195.

LI, Y. Deep reinforcement learning: an overview. *arXiv preprint arXiv:1701.07274*, 2018. DOI: 10.48550/arXiv.1701.07274.

LILICRAP, T. P.; HUNT, J. J.; PRITZEL, A.; HEESS, N.; EREZ, T.; TASSA, Y.; SILVER, D.; WIERSTRA, D. Continuous control with deep reinforcement learning. In: INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS, 4., 2016, San Juan. *Proceedings [...]*. San Juan: ICLR, 2016. DOI: 10.48550/arXiv.1509.02971.

LIN, X.; GROVLEN, A.; WERNER, K.; LI, J.; BALDEMAIR, R.; CHENG, J.-F. T.; PARKVALL, S.; LARSSON, D. C.; KOORAPATY, H.; FRENNE, M.; FALAHATI, S. 5G New Radio: unveiling the essentials of the next generation wireless access technology. *IEEE Communications Standards Magazine*, v. 3, n. 3, p. 30-37, set. 2019. DOI: 10.1109/MCOMSTD.001.1800036.

LIU, Y.; GUO, K.; WANG, Q. Fairness-centric resource allocation methods for eMBB and URLLC services in 5G and beyond networks. In: IEEE GLOBAL COMMUNICATIONS CONFERENCE (GLOBECOM), 2024, Cidade do Cabo. *Proceedings [...]*. Cidade do Cabo: IEEE, 2024. DOI: 10.1109/GLOBECOM52923.2024.10901576.

MAMANE, A.; EL GHAZI, M.; BARB, G.-R.; OTESTEANU, M. 5G heterogeneous networks: an overview on radio resource management scheduling schemes. In: MEDITERRANEAN CONGRESS OF TELECOMMUNICATIONS (CMT), 7., 2019, Fès. *Proceedings [...]*. Fès: IEEE, 2019. p. 1-5. DOI: 10.1109/CMT.2019.8931369.

MAMANE, A.; FATTAH, M.; EL GHAZI, M.; EL BEKKALI, M. Scheduling algorithms for 5G networks and beyond: classification and survey. *IEEE Access*, v. 10, p. 51643-51661, 2022. DOI: 10.1109/ACCESS.2022.3174579.

MNIH, V.; KAVUKCUOGLU, K.; SILVER, D. et al. Human-level control through deep reinforcement learning. *Nature*, v. 518, p. 529-533, 2015. DOI: 10.1038/nature14236.

MNIH, V.; KAVUKCUOGLU, K.; SILVER, D.; GRAVES, A.; ANTONOGLOU, I.; WIERSTRA, D.; RIEDMILLER, M. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. DOI: 10.48550/arXiv.1312.5602.

MÜLLER, M. K.; ADEMAJ, F.; DITTRICH, T.; FASTENBAUER, A.; RAMOS ELBAL, B.; NABAVI, A.; NAGEL, L.; SCHWARZ, S.; RUPP, M. Flexible multi-node simulation of cellular mobile communications: the Vienna 5G System Level Simulator. *EURASIP Journal on Wireless Communications and Networking*, v. 2018, n. 1, 227, 2018. DOI: 10.1186/s13638-018-1238-7.

NADERIALIZADEH, N.; SYDIR, J. J.; SIMSEK, M.; NIKOPOUR, H. Resource management in wireless networks via multi-agent deep reinforcement learning. *IEEE Transactions on Wireless Communications*, v. 20, n. 6, p. 3507-3523, jun. 2021. DOI: 10.1109/TWC.2021.3051163.

NGUYEN, H. T. T.; NGUYEN, M. T.; DO, H. T.; HUA, H. T.; NGUYEN, C. V. DRL-based intelligent resource allocation for diverse QoS in 5G and toward 6G vehicular networks: a comprehensive survey. *Wireless Communications and Mobile Computing*, v. 2021, 5051328, 2021. DOI: 10.1155/2021/5051328.

PANWAR, N.; SHARMA, S.; SINGH, A. K. A survey on 5G: the next generation of mobile communication. *Physical Communication*, v. 18, p. 64-84, 2016. DOI: 10.1016/j.phycom.2015.10.006.

POPOVSKI, P.; TRILLINGSGAARD, K. F.; SIMEONE, O.; DURISI, G. 5G wireless network slicing for eMBB, URLLC, and mMTC: a communication-theoretic view. *IEEE Access*, v. 6, p. 55765-55779, 2018. DOI: 10.1109/ACCESS.2018.2872781.

PRASAD, R.; SUNNY, A. QoS-aware scheduling in 5G wireless base stations. *IEEE/ACM Transactions on Networking*, v. 32, n. 3, p. 1999-2011, jun. 2024. DOI: 10.1109/TNET.2023.3342867.

PRATAP, A.; MISRA, R.; DAS, S. K. Maximizing fairness for resource allocation in heterogeneous 5G networks. *IEEE Transactions on Mobile Computing*, v. 20, n. 2, p. 603-619, 2021. DOI: 10.1109/TMC.2019.2948877.

QUEIROZ, G. A.; SILVA, É. R. da. A new channel and QoS aware scheduler algorithm for real-time and non-real-time traffic in 5G heterogeneous networks. *IEEE Latin America Transactions*, v. 22, n. 8, p. 659-669, ago. 2024. DOI: 10.1109/TLA.2024.10620389.

SEGUIN, M.; OMER, A.; KOOSHA, M.; MALANDRA, F.; MASTRONARDE, N. Deep reinforcement learning for downlink scheduling in 5G and beyond networks: a review. In: IEEE INTERNATIONAL SYMPOSIUM ON PERSONAL, INDOOR AND MOBILE RADIO COMMUNICATIONS (PIMRC), 2023, Toronto. *Proceedings [...]*. Toronto: IEEE, 2023. DOI: 10.1109/PIMRC56721.2023.10293754.

SHAHID, S. M.; KUMBI, G. G.; KIM, S.; KWON, S. QoS-aware downlink scheduling algorithm for GBR and non-GBR flows in 5G networks. In: INTERNATIONAL CONFERENCE ON ELECTRONICS, INFORMATION AND COMMUNICATION (ICEIC), 2026. *Proceedings [...]*. [S. l.]: IEEE, 2026. DOI: 10.1109/ICEIC69189.2026.11385931.

SHIYAHIN, A.; SCHWARZ, S.; RUPP, M. Quality of service aware scheduling in mixed traffic wireless networks. In: IEEE INTERNATIONAL WORKSHOP ON COMPUTER AIDED MODELING AND DESIGN OF COMMUNICATION LINKS AND NETWORKS (CAMAD), 2022. *Proceedings [...]*. [S. l.]: IEEE, 2022. DOI: 10.1109/CAMAD55695.2022.9966904.

SIDDIQUI, M. U. A.; QAMAR, F.; AHMED, F.; NGUYEN, Q. N.; HASSAN, R. Interference management in 5G and beyond network: requirements, challenges and future directions. *IEEE Access*, v. 9, p. 68932-68965, 2021. DOI: 10.1109/ACCESS.2021.3073543.

SILVER, D.; LEVER, G.; HEES, N.; DEGRIS, T.; WIERSTRA, D.; RIEDMILLER, M. Deterministic policy gradient algorithms. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 31., 2014, Pequim. *Proceedings [...]*. Pequim: PMLR, 2014. p. 387-395.

STIGENBERG, J.; SAXENA, V.; TAYAMON, S.; GHADIMI, E. QoS-aware scheduling in New Radio using deep reinforcement learning. In: IEEE INTERNATIONAL SYMPOSIUM ON PERSONAL, INDOOR AND MOBILE RADIO COMMUNICATIONS (PIMRC), 2021, Helsinki. *Proceedings [...]*. Helsinki: IEEE, 2021. p. 991-997. DOI: 10.1109/PIMRC50174.2021.9569581.

SUTTON, R. S.; BARTO, A. G. *Reinforcement learning: an introduction*. 2. ed. Cambridge: MIT Press, 2018.

VAN HASSELT, H.; GUEZ, A.; SILVER, D. Deep reinforcement learning with double Q-learning. In: AAAI CONFERENCE ON ARTIFICIAL INTELLIGENCE, 30., 2016, Phoenix. *Proceedings [...]*. Phoenix: AAAI Press, 2016. p. 2094-2100. DOI: 10.1609/aaai.v30i1.10295.

WANG, H.; BAI, Y.; XIE, X. Deep reinforcement learning based resource allocation in delay-tolerance-aware 5G industrial IoT systems. *IEEE Transactions on Communications*, v. 72, n. 1, p. 209-221, jan. 2024. DOI: 10.1109/TCOMM.2023.3322736.

WANG, Z.; SCHAUL, T.; HESSEL, M.; VAN HASSELT, H.; LANCTOT, M.; DE FREITAS, N. Dueling network architectures for deep reinforcement learning. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 33., 2016, Nova York. *Proceedings [...]*. Nova York: PMLR, 2016. p. 1995-2003.

WASSERSTEIN, R. L.; LAZAR, N. A. The ASA statement on p-values: context, process, and purpose. *The American Statistician*, v. 70, n. 2, p. 129-133, 2016. DOI: 10.1080/00031305.2016.1154108.

WATKINS, C. J. C. H. *Learning from delayed rewards*. 1989. Tese (Doutorado) - King's College, University of Cambridge, Cambridge, 1989.

WATKINS, C. J. C. H.; DAYAN, P. Q-learning. *Machine Learning*, v. 8, p. 279-292, 1992. DOI: 10.1007/BF00992698.

YU, H.; TALEB, T.; ZHANG, J. Deep reinforcement learning-based deterministic routing and scheduling for mixed-criticality flows. *IEEE Transactions on Industrial Informatics*, v. 19, n. 8, p. 8806-8816, ago. 2023. DOI: 10.1109/TII.2022.3222314.