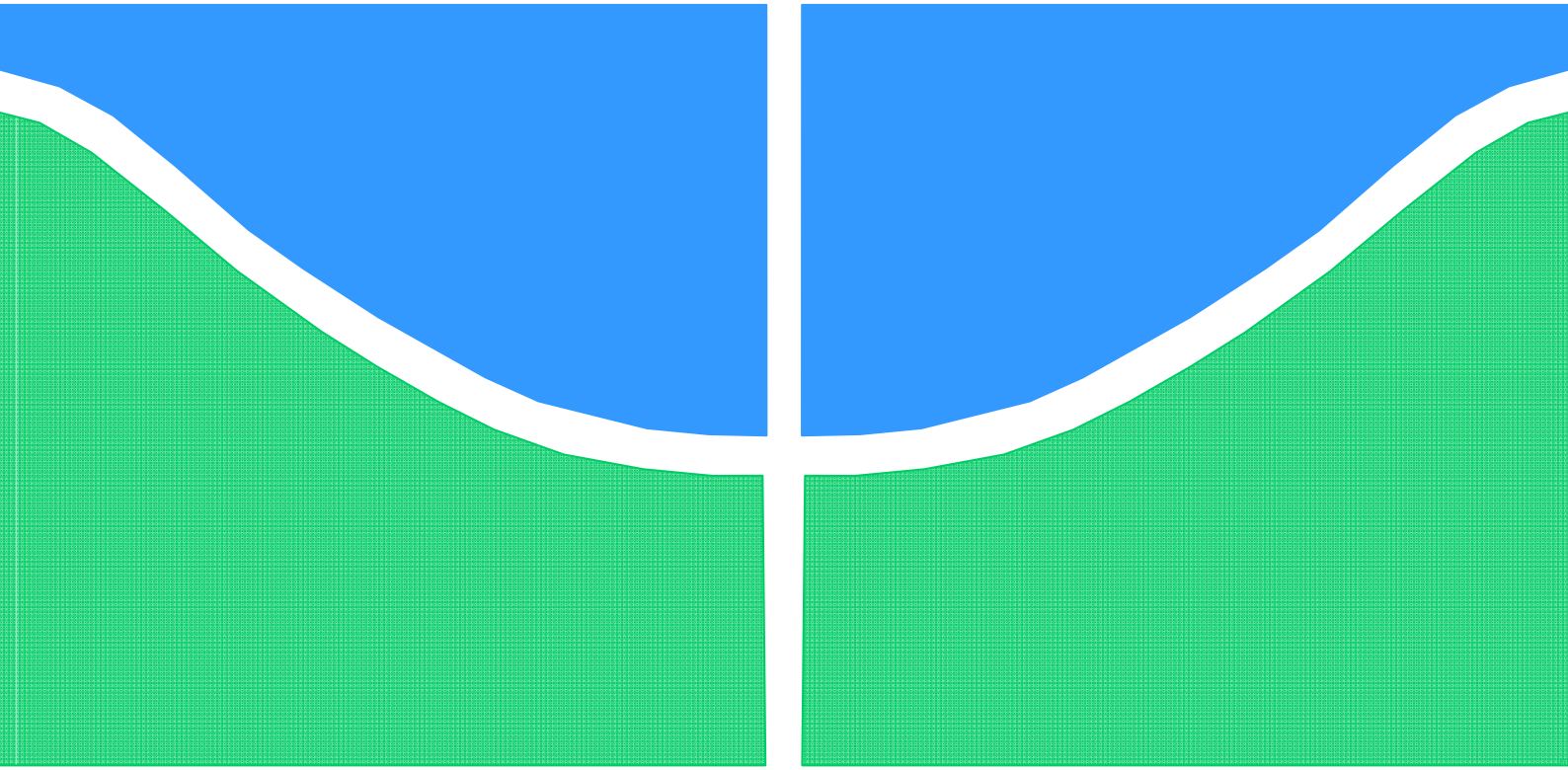


TESE DE DOUTORADO EM ENGENHARIA ELÉTRICA

Aplicação de Técnicas de Aprendizado por Reforço à Alocação de Recursos e ao Escalonamento de Usuários em Sistemas de Telecomunicações

João Paulo Leite

Brasília, maio de 2014

UNIVERSIDADE DE BRASILIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

APLICAÇÃO DE TÉCNICAS DE APRENDIZADO POR REFORÇO À ALOCAÇÃO DE RECURSOS E AO ESCALONAMENTO DE USUÁRIOS EM SISTEMAS DE TELECOMUNICAÇÕES

JOÃO PAULO LEITE

ORIENTADOR: PAULO HENRIQUE PORTELA DE CARVALHO

TESE DE DOUTORADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO: PPGENE.TD 087/2014
BRASÍLIA/DF: MAIO – 2014

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

APLICAÇÃO DE TÉCNICAS DE APRENDIZADO POR REFORÇO À ALOCAÇÃO DE RECURSOS E AO ESCALONAMENTO DE USUÁRIOS EM SISTEMAS DE TELECOMUNICAÇÕES

JOÃO PAULO LEITE

TESE SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISÍTOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR.

APROVADA POR:

_____

Prof. Paulo Henrique Portela de Carvalho, Dr., ENE/UnB
(Orientador)

_____

Prof. Alexandre Ricardo Soares Romariz, Dr., ENE/UnB
(Examinador Interno)

_____

Prof. Leonardo da Cunha Brito, Dr., UFG
(Examinador Externo)

_____

Prof. Robson Domingos Vieira, Dr., INdT
(Examinador Externo)

_____

Prof. Leonardo Aguayo, Dr., ENE/UnB
(Examinador Interno)

BRASÍLIA/DF, 30 DE MAIO DE 2014

FICHA CATALOGRÁFICA

LEITE, JOÃO PAULO
Aplicação de Técnicas de Aprendizado por Reforço à Alocação de Recursos e ao Escalonamento de Usuários em Sistemas de Telecomunicações [Distrito Federal] 2014.
xviii, 131p., 210 x 297 mm (ENE/FT/UnB, Doutor, Engenharia Elétrica, 2014)

Tese de Doutorado – Universidade de Brasília. Faculdade de Tecnologia.
Departamento de Engenharia Elétrica.
1.Aprendizado de Máquina 2.Aprendizado por Reforço
3.Escalonamento 4.Sistemas de Comunicação
I. ENE/FT/UnB II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

LEITE, J. P. (2014). Aplicação de Técnicas de Aprendizado por Reforço à Alocação de Recursos e ao Escalonamento de Usuários em Sistemas de Telecomunicações. Tese de Doutorado em Engenharia Elétrica, Publicação PPGENE.TD-087/2014, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 131p.

CESSÃO DE DIREITOS

AUTOR: João Paulo Leite.
TÍTULO: Aplicação de Técnicas de Aprendizado por Reforço à Alocação de Recursos e ao Escalonamento de Usuários em Sistemas de Telecomunicações.
GRAU: Doutor ANO: 2014

_____

João Paulo Leite
SMBS chácara 02 casa 01
71080-015, Brasília – DF.

Dedicatória

Ao meu avô, Jorge Ramalho (in memoriam).

João Paulo Leite

Agradecimentos

Ainda que corra o risco de esquecer alguns nomes, devo agradecer ...

Ao professor Dr. Paulo Henrique Portela de Carvalho, pela oportunidade de trabalho e pela paciência durante quase uma década de convivência.

Ao Dr. Robson Domingos Vieira, por apresentar novas formas de considerar os sistemas de comunicação.

Aos professores membros da banca examinadora, por aceitarem o convite para participação na defesa de tese e fornecerem sugestões que aprimoraram o trabalho.

Aos professores e funcionários do Departamento de Engenharia Elétrica (ENE) pelo convívio agradável e presteza no atendimento.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo suporte financeiro.

A Deus, por me permitir, a cada dia, novos desafios e oportunidades, e por colocadar em meu caminho pessoas verdadeiramente iluminadas, inspiradas e dispostas a fornecer o melhor de suas capacidades.

Aos que nunca deixaram de acreditar.

João Paulo Leite

RESUMO

A adaptação de enlace e o escalonamento de usuários são aspectos cruciais dos atuais sistemas de comunicação devido à demanda por alta eficiência espectral, de forma a se obter a maior vazão possível com base nos recursos espectrais disponíveis, e à grande variedade de aplicações de usuário, cada uma com diferentes requisitos de qualidade de serviço. A implantação tanto da adaptação de enlace quanto de algoritmos de seleção e escalonamento de usuários impõe certos desafios, pois as soluções atualmente utilizadas consideram modelos idealizados de terminais de transmissão e de recepção, bem como um canal de comunicação de natureza invariante e aplicações cujas exigências são imutáveis.

Nesse contexto, técnicas de aprendizado de máquina podem ser utilizadas como uma forma de superar as limitações impostas pelas técnicas tradicionais de modelagem e solução analítica dos problemas supracitados. Este trabalho apresenta como primeira contribuição uma solução para o problema de adaptação de enlace por modulação e codificação adaptativas em sistemas multiportadora utilizando técnicas de aprendizado por reforço por estados contínuos. Como segunda contribuição, ainda com respeito à adaptação de enlace, o trabalho propõe a utilização do aprendizado por reforço para a solução do problema de *bit loading* em sistemas multiportadora.

Como terceira contribuição, o trabalho propõe um algoritmo de seleção e escalonamento de usuários baseado na estratégia de aprendizado por reforço multi-objetivo, como uma forma de lidar com os diferentes requisitos de qualidade de serviço que são impostos pela heterogeneidade das aplicações que trafegam nas redes de comunicação atuais. Em particular, é considerado o problema de escalonamento de tráfego sensível ao atraso. Resultados de simulação mostram que as soluções propostas, baseadas em aprendizado por reforço, são capazes de explorar a variabilidade do meio de transmissão, de forma a suplantar as perdas que são introduzidas pela modelagem idealizada dos terminais de comunicação.

ABSTRACT

Link adaptation and scheduling are crucial aspects of communication systems since high spectral efficiency is required in order to obtain the highest throughput given the available spectrum resources and base stations should be able to service a wide range of quality of service requirements.

In this context, machine learning techniques can be used as a way to overcome the limitations imposed by traditional modeling techniques of the aforementioned problems. The first contribution of this thesis is to propose a solution to the problem of link adaptation for adaptive modulation and coding in multicarrier systems using a continuous-state reinforcement learning approach. As a second contribution, this thesis presents a solution to the bit loading problem in multicarrier systems by means of reinforcement learning.

As a third contribution, an algorithm for user selection and scheduling based on multi-objective reinforcement is proposed. In particular, the scheduling of delay-sensitive traffic is considered. Simulation results show that the proposed solutions, based on reinforcement learning, are able to exploit the variability of the transmission medium and overcome the losses that are introduced by idealized models of communication terminals and the communication channel.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

3GPP	*Third-Generation Partnership Project*
AMC	*Adaptive Modulation and Coding*
AWGN	*Additive White Gaussian Noise*
BER	*Bit Error Rate*
CRC	*Cyclic Redundancy Check*
CSI	*Channel State Information*
ET	*E-tightening*
EF	*Efficientizing*
FDD	*Frequency-Division Duplexing*
FIFO	*First-in, First-out*
FTP	*File Transfer Protocol*
kNN	*k-Nearest Neighbors*
LC	*Levin-Campello*
LQM	*Link Quality Metrics*
LSPI	*Least Squares Policy Iteration*
LSTD-Q	*Least-Square Temporal-Difference*
LTE	*Long Term Evolution*
MAP	*Maximum a Posteriori*
MDP	*Markov Decision Processes*
MIMO	*Multiple-input Multiple-output*
M-LWDF	*Modified Largest Weight Delay First*
M-QAM	*M-ary Quadrature Amplitude Modulation*
MT	*Maximum Throughput*
OFDM	*Orthogonal Frequency-division Multiplexing*
PER	*Packet Error Rate*
PF	*Proportional Fair*
QoS	*Quality of Servisse*
RF	Rádio frequência
RL	*Reinforcement Learning*
RR	*Round-Robin*
SARSA	*State-Action-Reward-State-Action*

SCM	*Spatial Channel Model*
SINR	*Signal-to-Interference-plus-noise Ratio*
SNR	*Signal-to-noise Ratio*
VoIP	*Voice over IP*

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Tradicionalmente, técnicas de inteligência artificial e de aprendizado de máquina não têm sido exploradas em sistemas digitais de comunicação. Como principais entraves à utilização das técnicas supracitadas, pode-se citar [1], em primeiro lugar, a simplicidade da abordagem estocástica utilizada, que tem como hipótese fundamental o problema de detecção de sinais em ruído branco do tipo gaussiano. Por ser capaz de fornecer *insights* para a predição do desempenho de sistemas digitais, métodos consagrados de estimação vêm sendo utilizados como base para projeto de sistemas de comunicação nos últimos 30 anos. Em segundo lugar, a complexidade de implementação e de tempo de processamento de alguns algoritmos de aprendizado de máquina representam uma limitação severa para sua difusão. Em terceiro lugar, a utilização de aprendizado de máquina requer o domínio do conhecimento de ambas as áreas para que a integração entre a modelagem tradicional e os paradigmas de inteligência artificial possam ser utilizados de forma efetiva.

Entretanto, esse cenário tem começado a mudar, ainda que de forma muito reservada, quando se passa a considerar a pesquisa recente em rádio cognitivo. Rádios cognitivos, conforme definidos por Mitola e Maguire [2], são rádios capazes de mudar seus parâmetros de transmissão com base na interação com o ambiente em que operam, e consistem em um novo modelo de operação na área de telecomunicações. Haykin [3] estende esta definição, considerando que o rádio cognitivo é um sistema de comunicação (sem fio) que observa o ambiente em que está inserido e é capaz de adaptar seu estado de funcionamento de acordo com as variações dos estímulos de RF (rádio frequência) aos quais está submetido. O estado de funcionamento deste rádio é modificado, em tempo real, por meio da alteração dos parâmetros de transmissão (potência de transmissão, frequência de operação, modulação utilizada etc.) com os objetivos de prover comunicação confiável entre as entidades envolvidas, além de buscar a utilização mais eficiente do espectro eletromagnético.

A partir da definição apresentada, pode-se inferir que as principais características do rádio cognitivo devem ser sua capacidade cognitiva, ou seja, a habilidade de capturar informações sobre o ambiente de RF, e sua reconfigurabilidade, isto é, a adaptação de seus parâmetros de transmissão e recepção às novas condições do canal de propagação.

A capacidade cognitiva, na literatura de rádio cognitivo, é representada por meio do ciclo cognitivo, dividido usualmente em três etapas [4]: o sensoriamento espectral, a análise do espectro e o gerenciamento do espectro. Na primeira fase do ciclo, o rádio busca

identificar as oportunidades de transmissão, ou lacunas espectrais, que são as faixas do espectro eletromagnético que se encontram subutilizadas. Em sua segunda etapa, são identificadas as características dos canais disponíveis, como nível de interferência e ganho do canal. Essas informações são combinadas na fase de gerenciamento, em que é tomada a decisão de qual canal será ocupado. De forma resumida, técnicas de processamento de sinais e de aprendizado de máquina podem ser utilizadas para prover sua capacidade cognitiva, enquanto que são as arquiteturas de rádio definido por *software* que permitem ao rádio cognitivo sua capacidade de reconfiguração.

Não restam dúvidas de que a literatura de rádio cognitivo é muito vasta, e as contribuições em seus diferentes ramos são diversas, tomando sempre como referência as habilidades do rádio cognitivo de observação, adaptação, raciocínio e aprendizagem. Apesar disso, a maioria dos trabalhos foca na etapa de sensoriamento espectral, isto é, a habilidade de observação, utilizando sobretudo abordagens analíticas [5, 4, 6, 7, 8, 9], o que fez com que, nos últimos anos, o sensoriamento espectral atingisse relativa maturidade [10].

Em número consideravelmente menor, há contribuições na parte de adaptação, seguido de raciocínio e aprendizagem (como exemplos, pode-se verificar as abordagens em [11, 12, 13]). Esta tendência mostra-se curiosa e, sob certo aspecto, surpreendente, uma vez que a idéia inicial de Mitola era a aplicação de técnicas de aprendizado de máquina, em especial as funções de raciocínio e aprendizagem, em sistemas de comunicação [2], sem excessiva concentração nas técnicas de sensoriamento espectral.

1.2 DEFINIÇÃO DO PROBLEMA

Antes da introdução do paradigma cognitivo em sistemas de comunicação, aprendizado e raciocínio não estavam presentes na arquitetura tradicional de transmissores e receptores. A observação e os mecanismos de adaptação eram governados e adotados por regras fixas e determinadas *a priori* no próprio *firmware* do terminal de comunicação. Com a introdução da capacidade de aprendizagem, pode-se vislumbrar as funções de um rádio inteligente conforme o *framework* mostrado na Fig. 1.1. O ambiente de rádio deve ser observado, e a informação adquirida por meio da etapa de observação é utilizada por algoritmos de aprendizado de máquina no processo de escolha dos parâmetros de transmissão utilizados, de forma a se atingir algum objetivo específico do sistema de comunicação, como a maximização da vazão ou a diminuição dos níveis de interfência.

Três vertentes podem ser identificadas quanto à utilização de algoritmos de aprendizado de máquina e inteligência artificial dentro do contexto de sistemas de comunicação e, mais especificamente, de rádio cognitivo: a utilização de técnicas heurísticas, de técnicas de aprendizado supervisionado e de técnicas de aprendizado não supervisionado.

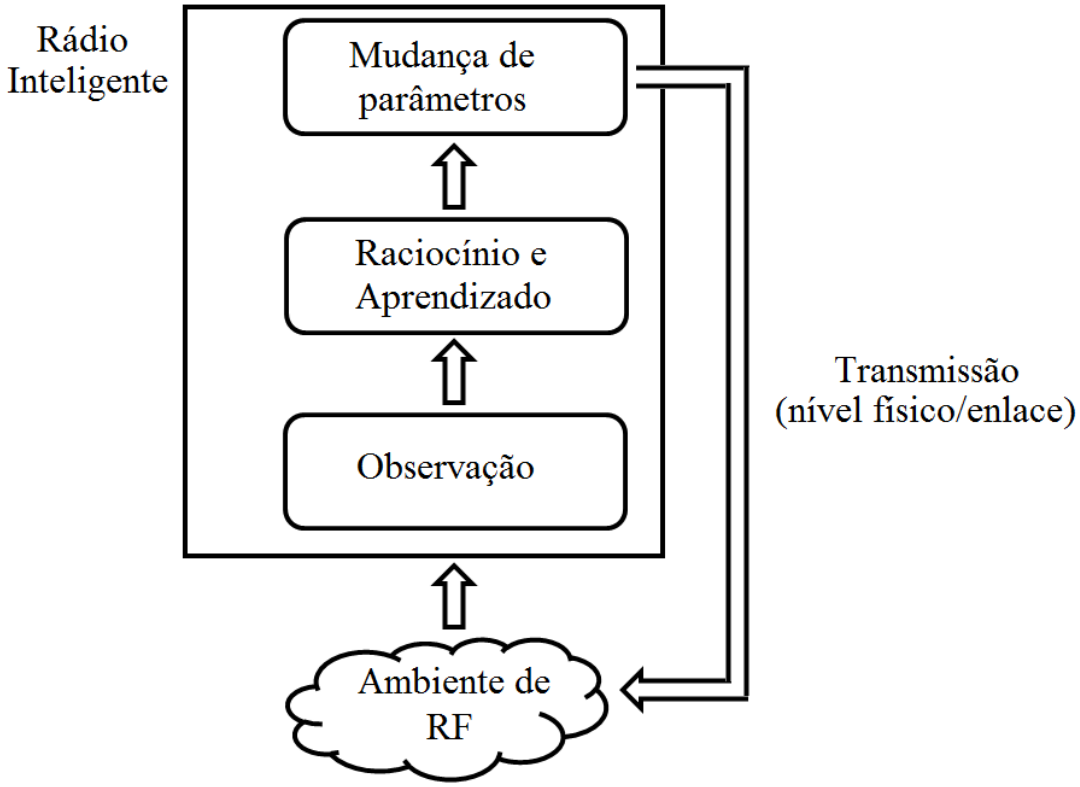


Figura 1.1: *Framework* que destaca as operações que devem ser executadas por um rádio inteligente.

Os métodos heurísticos, como algoritmos evolucionários e lógica nebulosa, conforme exposto em [14], são predominantemente utilizados para promover a otimização dos parâmetros de transmissão utilizados, tratando a reconfigurabilidade do rádio inteligente como um problema de otimização, não sendo dada enfase no processo de aprendizado. Em segundo lugar, encontram-se as técnicas de aprendizado supervisionado, como redes neurais artificiais [15, 16, 17, 18, 19] e máquinas de vetor de suporte [20, 21, 22], em que o rádio inteligente é treinado para reconhecer padrões de atuação de forma automática. Finalmente, há a utilização de técnicas de aprendizagem não supervisionada, como teoria dos jogos [23, 24, 25, 26] e algoritmos de aprendizado por reforço [27, 28, 29, 30], nas quais o rádio inteligente não dispõe de conjuntos de dados para seu treinamento prévio, sendo necessário obtê-los a partir de sua interação com o próprio ambiente em que opera.

Ainda que algoritmos de aprendizado de máquina sejam utilizados, o foco desses trabalhos é ainda a etapa de sensoriamento espectral, ou seja, a capacidade de observação do rádio inteligente. São tratados os problemas de detecção de recursos espectrais e compartilhamento espectral, mas pouca ou nenhuma ênfase é dada à reconfiguração de outros parâmetros de transmissão (além da frequência de operação utilizada), raciocínio ou aprendizagem. Logo, identifica-se um hiato de contribuições em outras funções cognitivas do rádio inteligente.

1.2.1 Adaptação de Enlace e Escalonamento

Devido à penetração de mercado e ao impacto que os sistemas de comunicação digital sem fio exercem na rotina de comunicação pessoal, entretenimento, comércio e indústria [1], é de fundamental importância que o projeto desses sistemas tenha como objetivos maximizar a taxa de transmissão de dados e sua robustez sob diferentes condições de operação.

Atualmente, os enlaces digitais de comunicação, independentemente da tecnologia utilizada, são projetados tendo em vista três conceitos fundamentais: capacidade, taxa de erro de *bit* e *overhead* de comunicação [1]. A capacidade diz respeito aos limites teóricos factíveis e aos limites práticos que podem ser alcançados por esses sistemas de comunicação. A taxa de erro de *bit* reflete a confiabilidade de um enlace e a qualidade de comunicação que pode ser provida. O *overhead* refere-se à codificação de canal e às informações de controle que devem trafegar pela rede de forma a viabilizar a comunicação.

Naturalmente, a maximização da taxa de transmissão e a robustez dos sistemas de comunicação estão relacionadas com os conceitos supracitados. A Fig. 1.2 mostra, de forma simplificada, os principais componentes de um enlace de comunicação. No transmissor, têm-se as funções de codificação de canal, modulação e alocação de potência. O sinal resultante é transmitido pelo canal de comunicação, que pode ser modelado por um filtro e uma componente aditiva de ruído, além de, possivelmente, uma componente de interferência. Finalmente, no receptor, é realizada a equalização do sinal recebido, além da demodulação e da decodificação de canal.

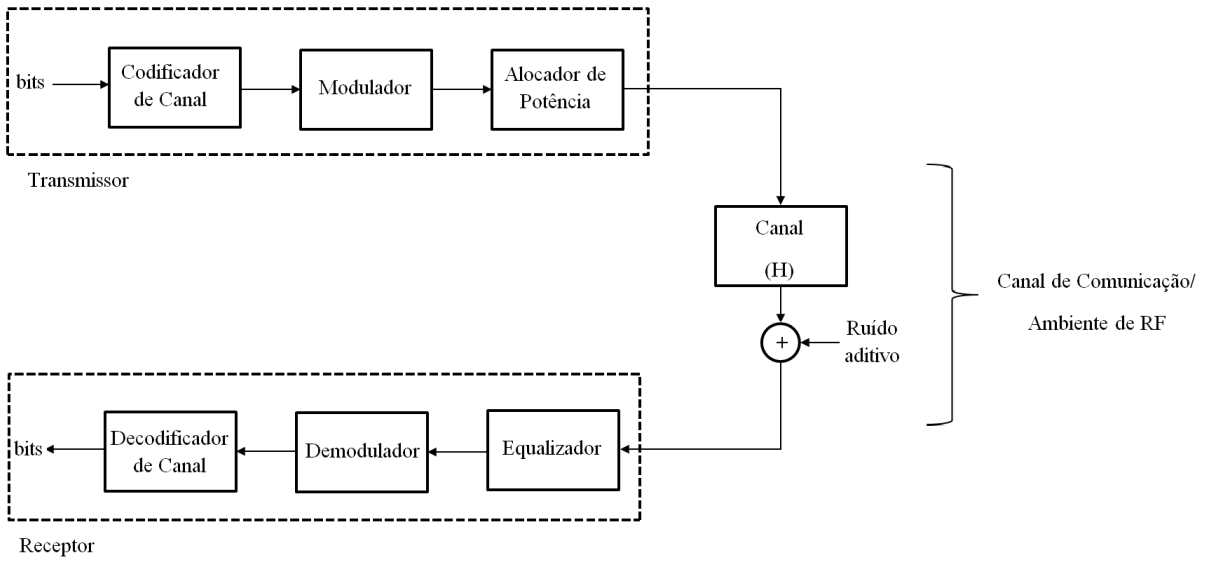


Figura 1.2: Diagrama simplificado dos elementos de um enlace de comunicação digital.

A capacidade teórica do enlace varia de acordo com as características do canal de comunicação [31]. Em sistemas práticos, observa-se diferença (*gap*) entre esta capacidade teórica e as taxas observáveis e alcançáveis (que são uma função da taxa de erro de *bit* e do *overhead* de

comunicação) para uma dada realização de canal. Esta diferença está relacionada aos parâmetros de transmissão utilizados, como modulação, codificação de canal e potência alocada.

Como o ganho do canal de comunicação varia com o tempo, sistemas de comunicação com parâmetros fixos de transmissão não são convenientes, pois acabam por subaproveitar os recursos de transmissão [31]. É sempre interessante buscar os melhores parâmetros de transmissão, de tal forma que se possa obter o menor *gap* de capacidade possível. Do contrário, o sistema irá operar com taxas de transmissão muito conservadoras, já que em geral ele foi projetado para a operação no pior caso. É possível ainda a falha completa do enlace de comunicação ao tentar operar com taxas de transmissão muito elevadas [32].

As técnicas de adaptação de enlace exploram os três conceitos fundamentais (capacidade, taxa de erro de *bit* e *overhead*) sobre os quais sistemas digitais são fundamentados e visam a seleção dinâmica dos parâmetros de transmissão com o objetivo de maximizar a taxa de transmissão, e constituem um elemento fundamental nos sistemas sem fio atuais, nos quais os recursos espectrais são limitados e taxas elevadas de transmissão de dados são requeridas.

Duas formas de se promover a adaptação de enlace são por meio da modulação e codificação adaptativas, que modificam os parâmetros utilizados pelo codificador e modulador, e da alocação de potência, que altera a potência de transmissão utilizada, conforme ilustrado na Fig. 1.3. Tradicionalmente, a adaptação de enlace é realizada utilizando-se abordagens analíticas, em que se buscam relações funcionais entre capacidade, modulação, codificação de canal e ganho de canal. Ainda que sejam capazes de fornecer *insights* úteis para o projetista sobre a operação dos sistemas digitais de comunicação, estas abordagens apresentam uma deficiência que merece considerável atenção: a dificuldade na derivação e obtenção de expressões fechadas que relacionem os diferentes parâmetros de transmissão, especialmente em sistemas OFDM (*orthogonal frequency-division multiplexing*) com codificação de canal, *interleaving* e múltiplas antenas. Apenas aproximações estão disponíveis, e a precisão que são capazes de fornecer são contestáveis [33, 34].

Além desses fatores, as derivações supõem que os transceptores utilizados são ideais e que o ruído introduzido pelo canal de comunicação é sempre branco e gaussiano. Entretanto, transceptores reais exibem comportamento distorcivo e não linear (devido à presença de osciladores, amplificadores, misturadores etc.), que não é modelado, e o canal de comunicação pode introduzir interferência colorida, impondo mais limitações à abordagem analítica e gerando soluções subótimas [35].

Outra observação ainda é necessária. Apesar de a maximização da taxa de transmissão consistir em um objetivo importante de sistemas de comunicação sem fio, esse objetivo não é único. Em um cenário de comunicação celular multi-usuário, tal como ilustrado na Fig. 1.4, ou até mesmo em um sistema de comunicação cognitivo com a presença de uma unidade centralizadora [36, 37, 38, 37], é necessário que a estação rádio-base atribua aos usuários os recursos de transmissão, de forma que todos possam obter acesso ao meio de comunicação para

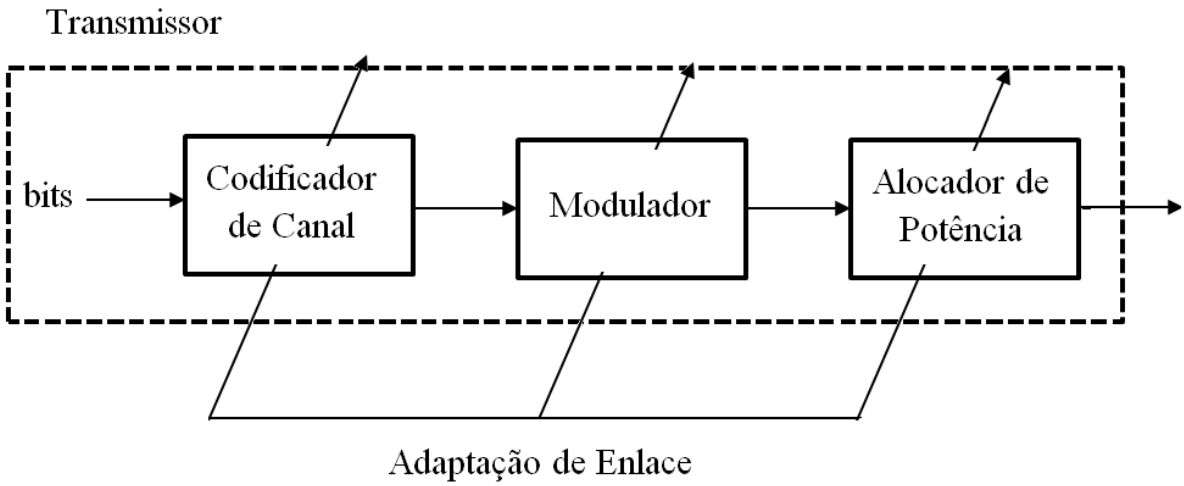


Figura 1.3: Blocos nos quais atua a estratégia de adaptação de enlace, com o objetico de otimizar os parâmetros de transmissão.

transmissão da informação [39].

Em linhas gerais, esse escalonamento de usuários é feito seguindo uma determinada política de alocação como, por exemplo, o aumento da vazão do sistema. Entretanto, como já citado anteriormente, este é apenas um dos objetivos a serem atendidos pois, se a estação rádio-base visa apenas a maximização de taxa de transmissão, então os recursos de transmissão serão atribuídos apenas aos usuários que apresentam as melhores condições de canal, prejudicando o acesso à rede dos usuários que se encontram mais distantes do nó central de transmissão, próximos à borda da célula. Claramente, esta abordagem gera um problema de justiça na distribuição dos recursos.

Deve-se considerar ainda o fato de que as aplicações que atualmente trafegam nas redes celulares são bastante heterogêneas, como VoIP (voz sobre IP), transmissão de vídeo em tempo real e navegação *web*. Naturalmente, essas aplicações não visam apenas maior taxa de transmissão; elas possuem diferentes requisitos de qualidade de serviço, como atraso de transmissão, largura de banda e perda de pacotes tolerada, o que gera desafios específicos para os algoritmos tradicionalmente utilizados para o escalonamento e seleção de usuários em redes celulares. Logo, infere-se que as estratégias de escalonamento de usuário devem também variar de acordo com o tipo de tráfego, devendo ser capazes de responder de forma eficiente à aplicações que sejam mais sensíveis ao atraso ou à taxa de erro de *bits* do sistema.

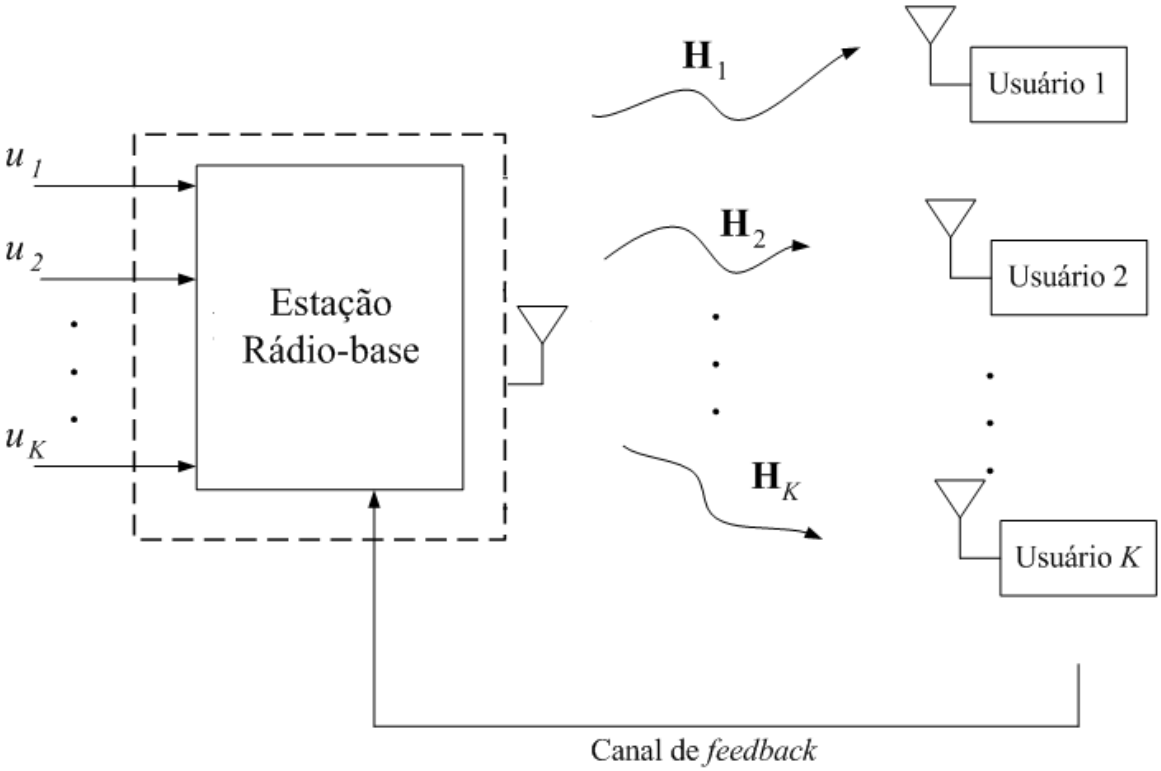


Figura 1.4: Diagrama de um sistema de comunicação multi-usuário celular.

1.3 OBJETIVOS

Este trabalho é dedicado a investigar os ganhos que a técnica de aprendizado por reforço pode trazer aos problemas de alocação de recursos e de escalonamento de usuários em sistemas de comunicação digital baseados em OFDM, abordando os problemas de adaptação de enlace (utilizando modulação e codificação adaptativas e alocação de potência) e de seleção e escalonamento de usuários. Diferentemente das outras abordagens na literatura, não será dada atenção à utilização de aprendizado de máquina para a solução do problema de sensoriamento espectral, mas sim na capacidade de aprendizagem e reconfiguração do rádio por meio de técnicas de inteligência artificial.

A escolha de sistemas OFDM com codificação se justifica pela importância que estes exercem nos atuais e possivelmente futuros padrões de camada física dos sistemas de comunicação, oferecendo uma solução viável para o tratamento de canais seletivos em frequência, além da dificuldade na obtenção de expressões analíticas fechadas para o desempenho desses sistemas sob diferentes condições de operação, o que faz com que as abordagens atuais para os problemas de alocação de recursos e de escalonamento de usuários ofereçam soluções sub-ótimas [40, 41, 42].

Como mostrado na Seção 1.2.1, as soluções de adaptação de enlace e escalonamento de usuários são basedas em aproximações analíticas que podem levar a resultados sub-ótimos pois,

em um cenário de comunicação real, uma série de hipóteses sobre as quais os modelos utilizados foram obtidos não são válidas, como é o caso do comportamento não ideal dos circuitos presentes nos tranceptores, ou a hipótese de interferência sendo modelada como ruído branco do tipo gaussiano, além de modelos bastante simplificados para o canal de comunicação. Nesse contexto, técnicas de aprendizado de máquina e inteligência artificial podem ser utilizadas como uma alternativa mais precisa e de melhor desempenho do que as soluções que são atualmente propostas para adaptação de enlace e escalonamento de usuários [1].

Dentro desse contexto, técnicas de aprendizado autônomo e não supervisionado de aprendizado são preferidas às técnicas de aprendizado supervisionado [43], sendo as candidatas mais apropriadas para a implementação de soluções viáveis de rádios reconfiguráveis, pois os terminais inteligentes devem ser capazes de aprender sem a influência de um tutor e em um ambiente de RF cujas características são potencialmente desconhecidas (níveis de interferência, número de usuários, comportamento do ruído, tráfego dos usuários etc.) e estão sujeitas a variações espaciais e temporais (devido à evolução da rede ou à variabilidade do meio de propagação). A própria natureza da tarefa a ser realizada, isto é, o mapeamento entre diversas situações possíveis do ambiente de RF e as ações a serem tomadas pelo rádio cognitivo, torna a utilização de técnicas supervisionadas uma tarefa consideravelmente complexa, especialmente no que concerne à obtenção de conjuntos de treinamento apropriados para os diferentes ambientes em que este rádio possa estar inserido.

Justifica-se então a escolha de algoritmos de aprendizado por reforço como um conjunto de soluções particularmente interessantes para promover a função de aprendizado do rádio pois permitem que, por um processo autônomo de tentativa e erro, o terminal possa explorar o ambiente, aprendendo a natureza subjacente deste, e que seja capaz ainda de refinar seu comportamento conforme interage com o ambiente, aprendendo o quão apropriada é uma ação para um dado estado do ambiente de rádio. Procura-se mostrar que o aprendizado por reforço, pode ser utilizado como uma alternativa mais precisa e de melhor desempenho do que as soluções que são atualmente propostas para adaptação de enlace e escalonamento de usuários. Conforme será abordado nos próximos capítulos, o aprendizado por reforço pode facilitar o processo de adaptação de enlace por implementações relativamente simples e flexíveis, sendo capaz de capturar a natureza complexa do canal de comunicação a partir de modelos que fazem poucas restrições quanto ao comportamento do enlace, sendo ainda capaz de se adaptar a diferentes variações de comportamento do canal de comunicação, sem a a necessidade de aproximações analíticas, que podem gerar soluções sub-ótimas ou conservadoras.

Logo, seguindo a linha apresentada, os problemas abordados serão três: a modulação e codificação adaptativas em sistemas OFDM, a alocação de *bits* e de potência (*bit loading*) em sistemas OFDM e o escalonamento de usuários em um cenário celular multi-usuário. Os dois primeiros tratam da melhor utiliziação do enlace de comunicação (em termos de eficiência espectral) aproveitando-se da variabilidade temporal do canal de comunicação, e o terceiro

trata da distribuição de recursos entre os usuários de uma mesma célula, sendo de fundamental importância para atender aos requisitos de qualidade de serviço das diferentes aplicações que trafegam em uma rede de comunicação celular. As estratégias são tratatas como problemas de aprendizado por reforço e, em cada caso, são também comparadas, por meio de simulações computacionais, com as abordagens mais comumente utilizadas na literatura. Busca-se ainda determinar cenários em que as soluções propostas sejam viáveis e capazes de providenciar melhor desempenho do que as soluções clássicas.

1.4 APRESENTAÇÃO DO MANUSCRITO

O texto desta tese está dividido em cinco capítulos. O primeiro é constituído pela presente introdução.

No capítulo 2, é apresentada a teoria fundamental da modelagem de aprendizado por reforço. Devido ao grande número de algoritmos de aprendizado por reforço, serão enfatizados apenas aqueles necessários à compreensão do presente trabalho.

O capítulo 3 trata do problema de modulação e codificação adaptativas. Nesse capítulo, é apresentada a formulação do problema, seu tratamento com o aprendizado por reforço e são realizadas simulações que comparam o desempenho da abordagem proposta com a solução por tabelas de consulta.

Em seguida, o capítulo 4 traz o problema de alocação de *bits* em sistemas OFDM. Assim como no capítulo anterior, busca-se apresentar o problema, sua solução tradicional, sua modelagem utilizando os conceitos de aprendizado por reforço e a comparação das soluções propostas.

O capítulo 5 introduz o escalonamento de usuários em sistemas OFDM e apresenta um *framework* para alocação de recursos em sistemas multiusuário utilizando técnicas de aprendizado por reforço multiobjetivo. São ainda apresentados resultados de simulação comparando o desempenho do *framework* proposto para alocação de recursos com os algoritmos e abordagens mais tradicionais encontradas na literatura.

Finalmente, o capítulo 6 apresenta as conclusões do trabalho e potenciais propostas de continuidade.

2 APRENDIZADO POR REFORÇO

2.1 INTRODUÇÃO

Este capítulo apresenta os principais conceitos envolvidos em problemas de aprendizado por reforço. O principal objetivo não é fornecer uma revisão abrangente de todas as vertentes de aprendizado por reforço, mas sim discutir os tópicos que são necessários à compreensão deste trabalho.

Inicialmente é fornecida uma visão geral do problema de aprendizado por reforço, e sua formalização matemática utilizando o conceito de processos de decisão Markovianos. Em seguida, são abordados os algoritmos utilizados para a solução de problemas de aprendizado por reforço. Consideram-se inicialmente as soluções baseadas em programação dinâmica, que dependem da modelagem completa do ambiente no qual o agente interage. Posteriormente são apresentados os algoritmos *Q-learning*, Diferenças Temporais e SARSA (*State-Action-Reward-State-Action*), nos quais a restrição quanto à modelagem do ambiente é retirada da formulação das soluções. É ainda dedicada uma seção ao tratamento de problemas de aprendizado por reforço por aproximação de funções, ou estados contínuos, nos quais o número de estados do ambiente é muito grande para ser representado de forma tabular. Por último, é mostrada uma vertente dos estudos de aprendizado por reforço multi-objetivo.

2.2 VISÃO GERAL

Nos sistemas baseados em aprendizado por reforço, há um agente que é capaz de aprender de maneira autônoma uma política ótima de atuação por meio de sua interação com o ambiente no qual está introduzido. O aprendizado do agente na abordagem de aprendizado por reforço é realizado por experimentação direta de ações sobre o ambiente e a observação da forma como este responde.

A Fig. 2.1 mostra a interação entre o agente e o ambiente, realizada de forma sequencial: o agente observa, em cada passo t da interação, o estado atual do ambiente s_t e, de acordo com este, escolhe uma ação a_t a realizar. Ao tomar esta ação, que é capaz de potencialmente alterar o estado em que se encontra o ambiente, o agente recebe um sinal de recompensa (ou reforço) r_t, que indica o quão apropriada é a ação tomada, levando o ambiente para o estado resultante s_{t+1}. O objetivo final do agente é determinar, após várias iterações, qual a melhor ação a ser executada em um determinado estado do ambiente, isto é, cabe determinar qual a melhor política de atuação.

As próximas seções visam formalizar matematicamente este processo de interação e aprendizado por meio da atuação do agente sobre o ambiente.

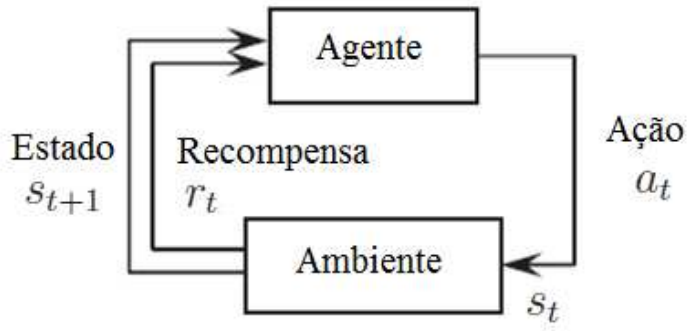


Figura 2.1: Diagrama de blocos que representa a interação do agente com o ambiente em um problema de aprendizado por reforço.

2.3 PROCESSOS DE DECISÃO MARKOVIANOS

Como considerado na seção anterior, o objetivo de um agente, em problemas de aprendizado por reforço, é de aprender uma política (ótima) de atuação que maximize a recompensa por ele recebida ao longo do período de interação com o ambiente. Formalmente, problemas de aprendizado por reforço são tradicionalmente tratados como processos de decisão Markovianos (MDP, ou *Markov Decision Processes*), que são processos de decisão sequenciais em que ações são tomadas em instantes de tempo bem definidos, denominados instantes de decisão.

Um processo de decisão Markoviano obedece à condição de Markov para a memória do sistema: o estado de um sistema no instante de tempo $t+1$ é determinado apenas pelo estado do sistema no instante t e da ação tomada pelo agente neste estado. Em outras palavras, o estado do sistema independe de sua história [44]. Dessa forma, permite-se que a decisão tomada dependa apenas do estado atual do ambiente.

Um processo de decisão Markoviano é definido formalmente por meio de uma quádrupla $(\mathcal{S}, \mathcal{A}, P, R)$, em que [45]:

- $\mathcal{S} = \{s_1, s_2, \ldots s_n\}$ é o conjunto finito de n estados do ambiente;
- $\mathcal{A} = \{a_1, a_2, \ldots a_m\}$ é o conjunto finito de m ações que o agente pode realizar;
- $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0; 1]$ é o modelo de transição Markoviano, em que $P(s, a, s')$ é a probabilidade de o ambiente ser levado ao estado $s' \in \mathcal{S}$ ao executar a ação $a \in \mathcal{A}$ no estado $s \in \mathcal{S}$;

- $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ é a função recompensa, em que $R(s, a, s')$ representa o reforço imediato ao agente pelo ambiente ao efetuar a transição do estado s para o estado s' ao selecionar a ação a.

É importante notar que a notação s_t representa o estado em que o ambiente se encontra no instante de decisão t, e que a_t representa a ação que é executada neste instante t. Naturalmente, $s_t \in \mathcal{S}$ e $a_t \in \mathcal{A}$. Considera-se ainda que $\mathcal{S}$ e $\mathcal{A}$ não variam com o tempo t, ou seja, são fixos para um dado problema.

É comum expressar a transição entre estados como $P(s, a, s') = \Pr(s_{t+1} = s' | s_t = s, a_t = a)$, em que s_{t+1} representa o estado do processo no instante $t + 1$, s_t o estado do processo no instante t e a_t a ação tomada ao observar o estado s_t. Em geral, considera-se que a cadeia de Markov que representa o ambiente é estacionária, de forma que não há dependência temporal de P ou R conforme previamente definidos [46].

O objetivo de um processo de decisão Markoviano é de determinar uma política $\pi : \mathcal{S} \rightarrow \mathcal{A}$ (que é um mapeamento entre estados e ações) que maximiza alguma função ao longo do tempo. A notação $\pi(s)$ indica qual ação deve ser tomada quando o ambiente se encontra no estado s.

No aprendizado por reforço, o agente deve aprender uma política ótima π^* que mapeia o estado atual s_t em uma ação desejada, de forma a maximizar uma recompensa acumulada.

2.4 PROGRAMAÇÃO DINÂMICA

Uma das formas de descrever a recompensa acumulada por um agente é por meio da função valor (cumulativo) esperado, ou função custo esperado, ou ainda função valor do estado, denotada por $V^{\pi}(s)$, que é gerada ao seguir uma política π a partir do estado s. Formalmente, ela é definida como [47]:

$$V^{\pi}(s) = \mathbb{E}\left\{ \sum_{t=0}^{\infty} \gamma^t r_t \, | s_{t=0} = s \right\} \tag{2.1}$$

em que r_t é o sinal recompensa recebido em t e $0 \leq \gamma \leq 1$ é um fator de desconto das recompensas futuras com respeito à recompensa imediata. O fator de desconto determina a importância relativa das recompensas futuras do agente com relação à recompensa atual. Um valor de γ próximo de zero faz com que o agente dê mais impotância às recompensas imediatas de suas ações (comportamento míope), ao passo que um valor de γ próximo da unidade faz com que o agente valorize igualmente as recompensas futuras (comportamento a longo prazo).

Uma política ótima estacionária π^* deve maximizar $V^{\pi}(s)$ para todos os estados, isto é,

$$\pi^* = \max_{\pi} V^{\pi}(s), \forall s \in \mathcal{S} \tag{2.2}$$

A política ótima é encontrada tradicionalmente por meio de técnicas de programação dinâmica, como os algoritmos de iteração de valor e de iteração de política, a serem descritos a seguir.

2.4.1 Princípio da Otimalidade de Bellman

Os algoritmos e técnicas de programação dinâmica que buscam a política ótima em um processo de decisão Markoviano são fundamentados no princípio da otimalidade de Bellman, que afirma que seguir uma política ótima entre um estado inicial e um estado final, passando por estados intermediários, é equivalente a seguir a melhor política entre o estado inicial e um dos estados intermediários, seguida da melhor política deste estado intermediário até o estado final. Matematicamente, pode-se escrever que, para um determinado problema, dada uma política ótima $\pi^* = \{a_0^*, a_1^*, a_2^*, \cdots, a_N^*\}$, então a política $\left\{a_t^*, a_{t+1}^*, a_{t+2}^*, \cdots, a_{t+N}^*\right\}$ também é ótima para o subproblema cujo estado inicial é s_t, com $0 < t < N$, sendo N o número de estados sequencialmente visitados [48].

Logo, para encontrar a política ótima de um sistema que está no estado s_t, é necessário encontrar a ação que leva ao melhor estado s_{t+1} e, a partir deste, seguir a política ótima até o estado final. Este fato pode ser demonstrado formalmente a partir das equações de Bellman, que consiste em uma definição recursiva da função valor do estado, dada pela Eq. (2.1):

$$V^{\pi}(s) = \mathbb{E}\left\{r_t + \gamma V^{\pi}(s_{t+1}) \,|s_{t=0} = s\right\} \tag{2.3}$$

de onde segue que, para a política ótima (definida pela Eq. (2.2)):

$$V^{*}(s) = \max_{\pi} \mathbb{E}\left\{r_t + \gamma V^{*}(s_{t+1}) \,|s_{t=0} = s\right\} \tag{2.4}$$

2.4.2 Iteração de Valor

No algoritmo de iteração de valor, a função valor ótima V^{π} é obtida por meio da iteração [49, 46]:

$$V(s) \leftarrow \max_{\pi(s)} \left[r(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} P(s, \pi(s), s') V'(s') \right] \tag{2.5}$$

em que

- s é o estado na iteração t do algoritmo;

- $\pi(s) = a$ é a ação realizada na iteração t;
- s' é o estado na iteração $t+1$, resultado de se tomar a ação a quando o ambiente se encontra em s;
- $V(s)$ é a função valor do estado s;
- V' é a função valor do estado estimada na iteração $t-1$.

O processo dado pela Eq. (2.5) é repetido até que $V(s) = V'(s)$. A política ótima é encontrada ao final do processamento por meio de:

$$\pi^*(s) \leftarrow \underset{\pi(s)}{\operatorname{argmax}} \left[r\left(s, \pi(s)\right) + \gamma \sum_{s' \in \mathcal{S}} P\left(s, \pi(s), s'\right) V^*\left(s'\right) \right] \tag{2.6}$$

2.4.3 Iteração de Política

No algoritmo de iteração de política, inicia-se o processo de aprendizado com uma política qualquer arbitrária π e calcula-se V^π resolvendo-se o sistema de equações dado por [49, 46]:

$$V^\pi(s) = r\left(s, a\right) + \gamma \sum_{s' \in \mathcal{S}} P\left(s, a, s'\right) V^\pi\left(s'\right) \tag{2.7}$$

Em seguida, determina-se uma nova política [49, 46]

$$\pi'(s) \leftarrow \underset{a}{\operatorname{argmax}} \left[r(s, a) + \gamma \sum_{s' \in \mathcal{S}} P\left(s, a, s'\right) V^\pi\left(s'\right) \right] \tag{2.8}$$

e

$$\pi \leftarrow \pi'(s) \tag{2.9}$$

É interessante notar que a Eq. (2.7) refere-se à avaliação de uma determinada política, e a Eq. (2.8) determina a melhora da política existente. As Eq. (2.7), (2.8) e (2.9) são repetidas até que se obtenha a convergência, ou seja, $V^\pi(s) = V^{\pi'}(s)$. Pelo princípio da otimalidade de Bellman, quando nenhuma melhora no valor de V for possível, então a política obtida é a política ótima [47].

2.4.4 Comentários

Os algoritmos baseados em programação dinâmica exibem algumas desvantagens: em primeiro lugar, a política obtida por meio da Eq. (2.8) é chamada de política gulosa (*greedy*),

pois baseia-se na ação que gera sempre o maior valor esperado. Entretanto, é possível que uma política inicial ruim indique uma ação que seja sub-ótima para o problema, e não ótima. Nesta situação, a solução do processo de decisão Markoviano fica limitada a um mínimo local. Na verdade, como pode ser inferido, a escolha da ação a pelo agente altera sua percepção e sua relação com o ambiente.

Uma forma de se contornar o problema é por meio de estratégias de exploração aleatórias. Nessas estratégias, o agente enfrenta o dilema conhecido como exploração versus explotação (*exploration vs. exploitation dilemma*). Na exploração, o agente deve reunir informações sobre o ambiente (tomando ações que não são necessariamente as ótimas para dados estados), e na explotação o agente busca utilizar as informações já conhecidas, maximizando o retorno esperado. Qualquer estratégia de solução deve busca um equilíbrio entre a exploração e a explotação do ambiente.

Uma segunda desvantagem dos algoritmos apresentados é seu custo computacional, tanto na computação das iterações dadas pela Eq. (2.5) ou da solução do sistema de equações dado pela Eq. (2.7).

Uma terceira desvantagem dos algoritmos apresentados é que eles dependem diretamente do conhecimento do modelo de transição e de recompensas do ambiente, ou seja, todos os valores de P e R. Em aplicações mais práticas do aprendizado por reforço, um modelo completo do ambiente não está disponível para o agente, sendo impossível a aplicação dos algoritmos apresentados. Dessa forma, são necessários algoritmos livres de modelo (*model-free*) para resolver esses problemas, nos quais não é necessária a modelagem completa do ambiente. Serão considerados como exemplos os algoritmos Q-Learning, SARSA e diferenças temporais.

2.5 Q-LEARNING

Uma outra forma de expressar a função valor do estado é escrevê-la em termos da função valor-ação, ou função Q. A função valor-ação indica o retorno esperado quando, em um estado s, toma-se a ação a e, a partir do próximo estado, volta-se a seguir a política π. Matematicamente, pode-se escrever que:

$$Q^{\pi}(s,a) = \mathbb{E}\left\{\sum_{t=0}^{\infty} \gamma^t r_t | s_{t=0} = s, a_{t=0} = a\right\} \tag{2.10}$$

O princípio do algoritmo *Q-Learning* é fazer com que o agente, em vez de maximizar $V^{\pi}(s)$, aprenda os valores da função valor-ação. Utilizando a condição de Markov, pode-se escrever:

$$
\begin{aligned}
Q^{\pi}(s,a) &= \mathbb{E}\left\{r_0 | s_{t=0}=s, a_{t=0}=a\right\} + \mathbb{E}\left\{\sum_{t=1}^{\infty}\gamma^t r_t | s_{t=1}=s', a_{t=1}=a'\right\} \\
&= \sum_{s'\in\mathcal{S}} P(s,a,s')\,R(s,a,s') + \gamma\mathbb{E}\left\{\sum_{t=0}^{\infty}\gamma^t r_{t+1} | s_{t=0}=s, a_{t=0}=a\right\} \\
&= \mathcal{R}(s,a) + \gamma\sum_{s'\in\mathcal{S}} P(s,a,s')\,Q^{\pi}(s',a'),
\end{aligned}
\tag{2.11}
$$

em que foi definido $\mathcal{R}(s,a) = P(s,a,s')\,R(s,a,s')$, a recompensa média tomando todos os possíveis estados futuros s' ao seguir a política π a partir das condições já consideradas. A Eq. (2.11) é conhecida como equação de Bellman, e indica que a função Q de um determinado par estado-ação pode ser expressa em termos da recompesa média do par estado-ação atual e da função Q do próximo par estado-ação.

A função Q ótima, $Q^*(s,a)$, é aquela que satisfaz a condição $Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a)$. Pelo princípio da otimalidade de Bellman, escreve-se, a partir da Eq. (2.11):

$$
Q^*(s,a) = \mathcal{R}(s,a) + \gamma\sum_{s'\in\mathcal{S}} P(s,a,s')\max_{a'\in\mathcal{A}} Q'(s',a') \tag{2.12}
$$

Como consequência, a Eq. (2.2) pode ser reescrita como

$$
\begin{aligned}
V^*(s) &= \max_{a\in\mathcal{A}} Q^*(s,a) \\
&= \max_{a\in\mathcal{A}}\left[\mathcal{R}(s,a) + \gamma\sum_{s'\in\mathcal{S}} P(s,a,s')\,V^*(s')\right]
\end{aligned}
\tag{2.13}
$$

Uma vez que os valores $Q^*(s,a)$ sejam conhecidos, a política ótima pode ser determinada diretamente, tomando-se, para cada estado, a ação que retorna o maior valor da função Q, isto é,

$$
\pi^*(s) = \operatorname*{argmax}_{a\in\mathcal{A}} Q^*(s,a) \tag{2.14}
$$

Seja então $\widetilde{Q}_t(s,a)$ a estimativa de $Q^*(s,a)$ em um dado instante t. O algoritmo *Q-Learning* aproxima iterativamente os valores de $Q^*(s,a)$, de tal forma que não é necessário o conhecimento explícito das probabilidades de transição conforme apresentadas na Eq. (2.11). A regra de atualização do algoritmo é [46]

$$
\widetilde{Q}_{t+1}(s,a) \leftarrow \widetilde{Q}_t(s,a) + \alpha\left[r(s,a) + \gamma\max_{a'}\widetilde{Q}_t(s',a') - \widetilde{Q}_t(s,a)\right] \tag{2.15}
$$

em que α é a taxa de aprendizagem, sujeita à restrição $0 < \alpha < 1$, e controla a convergência do

algoritmo. Em geral, faz-se a escolha

$$\alpha = \frac{1}{1 + v(s, a)} \tag{2.16}$$

em que $v(s, a)$ indica o número de visitas já realizadas ao estado s e tendo realizado a ação a [46].

Pode-se mostrar que, se o sistema pode ser modelado como um processo de decisão Markoviano, a função recompensa é limitada e as ações são escolhidas de tal forma que cada par estado-ação seja visitado um número infinito de vezes, então $\widetilde{Q}$ converge para Q^* com probabilidade 1. Outra propriedade notável do algoritmo é que as ações utilizadas no processo de aproximação da função Q podem ser escolhidas utilizando-se qualquer técnica de exploração e explotação. Por exemplo, a estratégia de exploração aleatória ϵ-*greedy* estabelece que o agente deve executar a ação com o maior valor de Q com probabilidade $1 - \epsilon$, ou escolher uma ação aleatória com probabilidade ϵ [47].

Matematicamente, a exploração ϵ-*greedy* pode ser expressa como:

$$\pi(s) = \begin{cases} a_{aleatoria}, & \text{com probabilidade } \epsilon \\ \text{argmax}_a \, \widetilde{Q}_t(s, a), & \text{com probabilidade } 1 - \epsilon \end{cases} \tag{2.17}$$

em que $a_{aleatoria}$ representa uma ação aleatória selecionada entre as ações possíveis a serem executadas no estado s.

Apesar de um dos critérios de convergência ser a visita de um estado um número infinito de vezes, na prática executa-se um número grande de visitas (de acordo com o contexto ou a complexidade da tarefa), e ainda assim observa-se a convergência do algoritmo [46].

2.6 DIFERENÇAS TEMPORAIS

O método das diferenças temporais foi proposto como uma versão adaptativa do algoritmo de iteração de valor, dado pela Eq. (2.5). Sua regra de atualização para a função valor esperado é dada por [47]:

$$\widetilde{V}^{\pi}_{t+1}(s) \leftarrow \widetilde{V}^{\pi}_{t}(s) + \alpha \left[r\left(s, a\right) + \gamma \widetilde{V}^{\pi}_{t}(s') - \widetilde{V}^{\pi}_{t}(s) \right] \tag{2.18}$$

em que $\widetilde{V}^{\pi}$ é a estimativa para o valor de V^{π}. Apesar da similaridade com o algoritmo de iteração de valor, a Eq. (2.18) não apresenta os termos referentes às probabilidades $P\left(s, a, s'\right)$. Estas são aprendidas de forma implícita a partir da iteração com o ambiente, sem qualquer conhecimento prévio, sendo uma combinação de métodos de programação dinâmica com simulação de Monte Carlo [47]. Mostra-se que, se a taxa de aprendizado α decair lentamente e a política π utilizada

for mantida fixa, então a Eq. (2.18) converge para $V^{\pi}(s)$, mas não necessariamente à política ótima.

2.7 SARSA

O algoritmo SARSA, diferentemente do *Q-Learning*, opera fazendo-se com que a política ótima seja aprendida em tempo de execução, estimando-se seu valor ao mesmo tempo que interage com o ambiente, sendo, portanto, um método *on-policy* (diferentemente do *Q-Learning*, que é *off-policy*). Uma vez que, a cada iteração, é estimado $\widetilde{Q}^{\pi}$ a partir de uma política inicial π, ao mesmo tempo em que é modificada a distribuição de probabilidades de escolha da próxima ação (a política utilizada).

Sua regra de atualização é dada por [47]:

$$\widetilde{Q}_{t+1}(s,a) \leftarrow \widetilde{Q}_t(s,a) + \alpha \left[r(s,a) + \gamma \widetilde{Q}_t(s',a') - \widetilde{Q}_t(s,a) \right] \tag{2.19}$$

Se a' for escolhido seguindo uma política gulosa, então o algoritmo SARSA se torna o próprio algoritmo *Q-Learning* [49, 46].

2.8 APRENDIZADO POR REFORÇO DE ESTADOS CONTÍNUOS

Da teoria exposta até o momento, percebe-se que, para que o problema de aprendizado por reforço seja resolvido por meio de estratégias iterativas, como é o caso dos algoritmos *Q-Learning*, diferenças temporais ou SARSA, é necessário que o agente possua armazenada em memória uma tabela que mapeia cada estado s em um valor da função valor do estado $V(s)$ (para o caso do algoritmo de diferenças temporais) ou uma tabela que mapeia cada par estado-ação (s,a) em um respectivo valor de função valor-ação $Q(s,a)$. Em outras palavras, a aplicabilidade dos algoritmos apresentados depende fortemente da representação tabular dos pares estado-ação e das políticas utilizadas.

Esta característica pode se tornar uma limitação muito séria para a implementação prática desses algoritmos. Dependendo da estrtutura do problema tratado, os conjuntos utilizados para a representação em estados do ambiente ou das ações que podem ser tomadas pelo agente podem possuir um valor muito grande de elementos, da ordem de centenas de milhões, o que torna o custo de armazenamento das tabelas muito elevado em termos da memória computacional requerida, e pode tornar a convergência dos algoritmos extremamente lenta, sendo esta uma característica indesejável para aprendizado em tempo real.

Portanto, dependendo do tipo de problema, a representação exata de todos os pares estado-ação e das políticas seguidas não é possível pois o número de estados é extremamente elevado (ou, por vezes, potencialmente infinito), sendo esta característica conhecida como o problema da dimensionalidade (*curse of dimensionality*) [50].

Nos casos em que a representação sofre com o problema da dimensionalidade, a representação tabular exata da função $Q^{\pi}(s,a)$, sendo impossível ou muito custosa computacionalmente, é substituída por uma aproximação parametrizada $Q^{\pi}(s,a;\mathbf{w})$, em que $\mathbf{w}$ são os parâmetros ajustáveis do aproximador. Neste caso, é necessário o armazenamento em memória apenas dos parâmetros utilizados pela aproximação, além da própria arquitetura de aproximação utilizada [50].

A teoria apresentada nesta seção, assim como os algoritmos utilizados, possuem como base a discussão apresentada em [45].

Conforme abordado anteriormente, é sabido que o valor exato da função Q^{π} para todos os pares estado-ação pode ser encontrado resolvendo-se o sistema linear de equações

$$Q^{\pi}(s,a) = \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s,a,s')\, Q^{\pi}(s',a') \tag{2.20}$$

A Eq. (2.20) pode ser expressa na forma matricial como

$$\mathbf{Q}^{\pi} = \boldsymbol{\mathcal{R}} + \gamma \mathbf{P} \boldsymbol{\Pi}_{\pi} \mathbf{Q}^{\pi} \tag{2.21}$$

em que $\mathbf{Q}^{\pi}$ e $\boldsymbol{\mathcal{R}}$ são vetores de tamanho $|\mathcal{S}|\,|\mathcal{A}|$, e $\mathbf{P}$ é uma matriz de dimensão $|\mathcal{S}|\,|\mathcal{A}| \times |\mathcal{S}|$ que contém as probabilidades de transição do modelo do ambiente, ou seja,

$$\mathbf{P}((s,a),s') = P(s,a,s') \tag{2.22}$$

e $\boldsymbol{\Pi}_{\pi}$ é uma matriz $|\mathcal{S}| \times |\mathcal{S}|\,|\mathcal{A}|$ que descreve a política π utilizada,

$$\boldsymbol{\Pi}_{\pi}(s',(s',a')) = \pi(a',s') \tag{2.23}$$

em que $\pi(a,s)$ denota a probabilidade de tomar a ação a quando o ambiente se encontra no estado s. Para o caso de uma política determinística, os únicos valores que $\pi(a,s)$ pode apresentar são 0 ou 1.

Define-se ainda o operador de Bellman T_{π} sobre $Q(s,a)$ como:

$$T_{\pi}[Q(s,a)] = \mathcal{R}(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s,a,s')\, Q^{\pi}(s',a') \tag{2.24}$$

Mostra-se que Q^π é ponto fixo do operador de Bellman [51], ou seja, para qualquer valor inicial de Q, aplicações sucessivas de T_π sobre Q fazem com que Q convirja para Q^π.

2.8.1 Arquitetura Linear de Aproximação de Funções

Uma forma comum de aproximação de funções é por meio de uma arquitetura linear, em que a função de interesse é parametrizada utilizando-se uma combinção linear de k funções de base [46, 52]:

$$\begin{aligned} Q^\pi(s,a\,;\mathbf{w}) &= \sum_{l=1}^{k} \phi_l(s,a)\, w_l \\ &= \begin{bmatrix} \phi_1(s,a) & \cdots & \phi_k(s,a) \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_k \end{bmatrix} \\ &= \boldsymbol{\phi}^T(s,a) \cdot \mathbf{w} \end{aligned} \tag{2.25}$$

As funções de base ϕ_l são escolhidas *a priori* e mantidas fixas para um dado problema e são, em geral, uma função não linear de s e a.

Seja $\mathbf{Q}^\pi$ a função valor (desconhecida) de uma política π, dada por um vetor coluna de $|\mathcal{S}|\,|\mathcal{A}|$ elementos. Seja ainda $\widehat{\mathbf{Q}}^\pi$ um vetor de aproximações da função $\mathbf{Q}^\pi$ obtido por meio da aproximação linear deste. Definindo

$$\phi(s,a) = \begin{bmatrix} \phi_1(s,a) \\ \vdots \\ \phi_k(s,a) \end{bmatrix} \tag{2.26}$$

$\widehat{\mathbf{Q}}^\pi$ pode ser expresso de forma compacta como

$$\widehat{\mathbf{Q}}^\pi = \mathbf{\Phi}\mathbf{w} \tag{2.27}$$

em que $\mathbf{\Phi}$ é uma matriz $|\mathcal{S}|\,|\mathcal{A}| \times k$ na forma

$$\mathbf{\Phi} = \begin{bmatrix} \phi(s_1,a_1)^T \\ \vdots \\ \phi(s,a)^T \\ \vdots \\ \phi_k\left(s_{|\mathcal{S}|}, a_{|\mathcal{A}|}\right) \end{bmatrix} \tag{2.28}$$

Cada linha de $\mathbf{\Phi}$ contém o valor das funções de base para um dado par (s, a), e cada coluna de $\mathbf{\Phi}$ contém o valor de uma dada função base para todos os pares (s, a).

O objetivo é encontrar um método de avaliação de políticas e de projeções de tal forma que, dadas uma política π e um modelo de ambiente, sejam obtidos os parâmetros $\mathbf{w}$ de tal forma que $\widehat{\mathbf{Q}}^{\pi}$ seja uma boa aproximação para $\mathbf{Q}^{\pi}$.

Uma definição natural para o conceito de boa aproximação é que a aproximação da função Q^{π} também satisfaça a equação de Bellman. Então, substituindo $\widehat{\mathbf{Q}}^{\pi}$ no lugar de $\mathbf{Q}^{\pi}$ na Eq. (2.21), tem-se

$$\widehat{\mathbf{Q}}^{\pi} \approx \boldsymbol{\mathcal{R}} + \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \widehat{\mathbf{Q}}^{\pi} \tag{2.29}$$

$$\mathbf{\Phi w} \approx \boldsymbol{\mathcal{R}} + \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi w} \tag{2.30}$$

$$\left(\mathbf{\Phi} - \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi}\right) \mathbf{w} \approx \boldsymbol{\mathcal{R}} \tag{2.31}$$

Este sistema linear sobredeterminado pode ser resolvido no sentido dos mínimos quadrados, obtendo-se

$$\mathbf{w} = \left(\left(\mathbf{\Phi} - \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi}\right)^{T} \left(\mathbf{\Phi} - \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi}\right)\right)^{-1} \left(\mathbf{\Phi} - \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi}\right)^{T} \boldsymbol{\mathcal{R}} \tag{2.32}$$

que é conhecida como a aproximação de Bellman que minimiza o erro residual entre $\widehat{\mathbf{Q}}^{\pi}$ e $\mathbf{Q}^{\pi}$.

Uma outra forma de obter uma aproximação para $\mathbf{Q}^{\pi}$ é impor a condição de que a aproximação $\widehat{\mathbf{Q}}^{\pi}$ seja também um ponto fixo do operador de Bellman, ou seja,

$$T_{\pi} \widehat{\mathbf{Q}}^{\pi} \approx \widehat{\mathbf{Q}}^{\pi} \tag{2.33}$$

Dessa forma, pode-se escrever que

$$\begin{aligned} \widehat{Q}^{\pi} &= \mathbf{\Phi} \left(\mathbf{\Phi}^{T} \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^{T} \left(T_{\pi} \widehat{\mathbf{Q}}^{\pi}\right) \\ &= \mathbf{\Phi} \left(\mathbf{\Phi}^{T} \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^{T} \left(\boldsymbol{\mathcal{R}} + \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \widehat{\mathbf{Q}}^{\pi}\right) \end{aligned} \tag{2.34}$$

Por meio da Eq. (2.27), mostra-se que a solução desse sistema é dada por:

$$\mathbf{w} = \left(\mathbf{\Phi}^{T} \left(\mathbf{\Phi} - \gamma \mathbf{P} \mathbf{\Pi}_{\pi} \mathbf{\Phi}\right)\right)^{-1} \mathbf{\Phi}^{T} \boldsymbol{\mathcal{R}} \tag{2.35}$$

Experimentalmente, observa-se que a solução dada pela Eq. (2.35) fornece aproximações

melhores e mais estáveis do que a solução dada pela Eq. (2.32). Logo, os algoritmos que serão apresentados levam em consideração a manipulação da forma dada pela Eq. (2.35) para a aproximação da função Q, em que não é necessário o conhecimento das matrizes $\mathbf{P}$ e $\boldsymbol{\mathcal{R}}$, ou seja, não é necessário que o agente conheça a modelagem completa do ambiente.

A seguir, nas Seções 2.8.2 e 2.8.3, serão apresentandos dois dos algoritmos que são utilizados para a solução do problema de aprendizado por reforço de estados contínuos.

2.8.2 LSTD-Q

O algoritmo LSTD-Q (*Least-Square Temporal-Difference*) utiliza como base a Eq. (2.34) para determinar uma aproximação $\widehat{\mathbf{Q}}^{\pi}$ para $\mathbf{Q}^{\pi}$ de uma política π a partir de um conjunto finito D de L observações do ambiente, na forma

$$D = \{(s_i, a_i, r_i, s'_i) \,|i = 1, 2, \ldots, L\} \tag{2.36}$$

Supondo que há k funções de base na arquitetura de aproximação linear, este problema de aproximação é equivalente a aprender os parâmetros $\mathbf{w}$ de $\widehat{\mathbf{Q}}^{\pi} = \boldsymbol{\Phi}\mathbf{w}$. Utilizando a Eq. (2.34), mostra-se que $\mathbf{w}$ pode ser obtido a partir da solução de um sistema linear $k \times k$ na forma [45]:

$$\mathbf{A}\mathbf{w} = \mathbf{b} \tag{2.37}$$

em que

$$\mathbf{A} = \boldsymbol{\Phi}^T \left(\boldsymbol{\Phi} - \gamma \mathbf{P}\boldsymbol{\Pi}_{\pi}\boldsymbol{\Phi}\right) \tag{2.38}$$

e

$$\mathbf{b} = \boldsymbol{\Phi}^T \boldsymbol{\mathcal{R}} \tag{2.39}$$

As matrizes $\mathbf{A}$ e $\mathbf{b}$ não podem ser determinadas *a priori* pois, em geral, $\mathbf{P}$ e $\boldsymbol{\mathcal{R}}$ são desconhecidos ou apresentam muitos elementos, de forma que sua utilização direta não é computacionalmente viável. Entretanto, $\mathbf{A}$ e $\mathbf{b}$ podem ser apreendidas usando as amostras do conjunto D apresentado na Eq. (2.36).

Expandindo as formas de $\mathbf{A}$ e $\mathbf{b}$, obtém-se:

$$\begin{aligned} \mathbf{A} &= \boldsymbol{\Phi}^T \left(\boldsymbol{\Phi} - \gamma \mathbf{P}\boldsymbol{\Pi}_{\pi}\boldsymbol{\Phi}\right) \\ &= \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} P\left(s, a, s'\right) \left[\phi(s, a) \left(\phi(s, a) - \gamma\phi(s', \pi(s'))\right)^T\right] \end{aligned} \tag{2.40}$$

$$\begin{aligned}\mathbf{b} &= \mathbf{\Phi}^T \boldsymbol{\mathcal{R}} \\ &= \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} \sum_{s'\in\mathcal{S}} P\left(s,a,s'\right)\left[\phi(s,a)R(s,a,s')\right]\end{aligned} \tag{2.41}$$

As Eq. (2.40) e (2.41) mostram que **A** e **b** possuem uma estrutura especial que pode ser explorada. A matriz **A** consiste na soma de termos na forma

$$\phi(s,a)\left(\phi(s,a)-\gamma\phi(s',\pi(s'))\right)^T \tag{2.42}$$

ao passo que o vetor **b** consiste na soma de termos do tipo

$$\phi(s,a)R(s,a,s') \tag{2.43}$$

Cabe notar que os somatórios são tomados sobre o espaço de s, a e s', e ponderados pelas probabilidades de transição $P\left(s,a,s'\right)$. Para problemas que apresentam grande dimensionalidade em s e a, não é prático realizar este cômputo para todas as combinações de (s,a,s'). Entretanto, é possível conhecer algumas realizações de (s,a,s') já que, se o processo de amostragem não é viesado, s' deve ser originário da distribuição $P\left(s,a,s'\right)$.

Dessa forma, pode-se fazer

$$\widetilde{\mathbf{A}} = \frac{1}{L}\sum_{i=1}^{L}\left[\boldsymbol{\phi}\left(s_i,a_i\right)\boldsymbol{\phi}^T\left(s_i,a_i\right)-\gamma\phi\left(s_i,a_i\right)\boldsymbol{\phi}^T\left(s_i',\pi\left(s_i'\right)\right)\right] \tag{2.44}$$

$$\widetilde{\mathbf{b}} = \frac{1}{L}\sum_{i=1}^{L}\boldsymbol{\phi}\left(s_i,a_i\right)r_i \tag{2.45}$$

No limite, conhecendo várias combinações de (s,a,s') e somando os termos correspondentes, é possível obter as aproximações desejadas. Matematicamente, temos:

$$\lim_{L\to\infty}\widetilde{\mathbf{A}} = \mathbf{\Phi}^T\left(\mathbf{\Phi}-\gamma\mathbf{P\Pi}_\pi\mathbf{\Phi}\right) \tag{2.46}$$

$$\lim_{L\to\infty}\widetilde{\mathbf{b}} = \mathbf{\Phi}^T\boldsymbol{\mathcal{R}} \tag{2.47}$$

A aproximação fornecida pelas Eq. (2.44) e (2.45) utiliza o mesmo princípio da iteração mostrada na Eq. (2.18), pois utiliza-se o fato de que o número de ocorrências de (s,a,s') será proporcional à $P\left(s,a,s'\right)$

A operação do LSTD-Q está resumida no Algoritmo 1.

2.8.3 LSPI

Cabe notar que o algoritmo LSTD-Q é capaz de aprender a aproximação $\widehat{Q}^{\pi}$ para uma dada política π a partir de amostras do conjunto D, ou seja, é capaz de avaliar políticas (*policy evaluation*). Em problemas de aprendizado por reforço, é necessário ainda encontrar a política ótima π^*.

O algoritmo LSPI (*Least Squares Policy Iteration*) possui como objetivo aprender uma política (e, em particular, a política ótima) a partir de um conjunto D de amostras, utilizando o algoritmo LSTD-Q em sua formulação [45]. O algoritmo está apresentado em Algoritmo 2.

A política ótima é a política gulosa sobre uma função de valor ótima, e pode ser determinada a partir de

$$\pi^*(s) = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, \widehat{Q}(s,a) = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, \phi(s,a)^T \mathbf{w} \tag{2.48}$$

Algoritmo 1 *LSTD-Q* (D, k, ϕ, γ, π)

1. $\widetilde{\mathbf{A}} \leftarrow \mathbf{0}$
2. $\widetilde{\mathbf{b}} \leftarrow \mathbf{0}$
3. Para cada $(s, a, r, s') \in D$
   4. $\widetilde{\mathbf{A}} \leftarrow \widetilde{\mathbf{A}} + \phi(s,a)\left(\phi(s,a) - \gamma\phi(s', \pi(s'))\right)$
   5. $\widetilde{\mathbf{b}} \leftarrow \widetilde{\mathbf{b}} + \phi(s,a)r$
6. $\mathbf{w} \leftarrow \widetilde{\mathbf{A}}^{-1}\widetilde{\mathbf{b}}$

Algoritmo 2 *LSPI* (D, k, ϕ, γ, π_0)

1. $\pi' \leftarrow \pi_0$
2. Repita:
   3. $\pi \leftarrow \pi'$
   4. $\pi' \leftarrow$ LSTD-Q (D, k, ϕ, γ, π)
5. até que $\pi \approx \pi'$

2.9 APRENDIZADO POR REFORÇO MULTI-OBJETIVO

Conforme apresentado ao longo deste capítulo, o aprendizado por reforço é fundamentado nos processos de decisão sequenciais do tipo Markoviano e, portanto, sua solução está condicionada a uma série de hipóteses relativamente fortes e restritivas. Conforme abordado na Seção 2.3, a

formalização requer um número finito de estados que o ambiente pode assumir e de ações que o agente pode executar, requer o conhecimento do modelo de transição Markoviano do ambiente e exige o conhecimento da função recompensa que exige apenas um critério de maximização.

Aos poucos, essas premissas podem ser relaxadas de modo a obter uma classe maior de problemas que podem ser resolvidos ainda por meio de técnicas de aprendizado por reforço. Por exemplo, na seção 2.5 foi mostrado o algoritmo iterativo *Q-Learning*, que não exige o conhecimento das probabilidades de transição do modelo do ambiente, sendo estas aprendidas de forma implícita.

Em seguida, na seção 2.8, considerou-se o problema da dimensionalidade na representação tabular utilizada pelos algoritmos iterativos, e a hipótese de um conjunto finito e discreto de estados foi enfraquecida para uma representação contínua por meio da teoria de aproximação de funções.

Outra premissa que pode ser relaxada é que a função recompensa seja aditiva e representada por um escalar, refletindo apenas um critério de maximização. Na verdade, vários problemas de interesse prático são modelados com funções recompensa multidimensionais, representando diferentes critérios a serem otimizados. Esta seção apresenta parte do formalismo necessário para o tratamento dessa classe de problemas, denominados de aprendizado por reforço multi-objetivo, e baseia-se principalmente na técnica fornecida em [53].

Conforme apresentado anteriormente, no aprendizado por reforço, um agente interage com o ambiente de forma a aprender um comportamento (política) ótimo, no sentido de maximizar valores acumulados de recompensa. A maior parte das técnicas de aprendizado por reforço é baseada em um sinal de recompensa escalar, ou seja, busca-se otimizar uma função objetivo unidimensional [54].

Uma extensão natural das técnicas tradicionais de aprendizado por reforço é considerar o caso em que há dois ou mais objetivos a serem atingidos por um agente, sendo cada um desses objetivos descritos por meio de um sinal recompensa, existindo, portanto, múltiplos sinais de recompensa. Considerar-se-á o caso em que um agente, ao atuar sobre o ambiente, recebe um sinal de recompensa $\vec{r}(s,a)$ que possui várias componentes, ou seja, é um vetor na forma

$$\vec{r}(s,a) = \begin{bmatrix} r_1(s,a) & r_2(s,a) & \cdots & r_N(s,a) \end{bmatrix} \tag{2.49}$$

em que N indica o número de objetivos a serem atendidos ou, de forma equivalente, o número de dimensões da recompensa recebida. O aprendizado de uma política ótima em várias situações depende da capacidade do agente aprender na presença de mais de um sinal de reforço

A seguir, será considerada uma breve descrição das abordagens consideradas para a solução do problema de aprendizado por reforço multi-objetivo (a citar, os algoritmos baseados em política única e os algoritmos baseados em múltiplas políticas), sendo enfatizado o algortimo de iteração

no fecho convexo [53].

Entretanto, antes de prosseguir, cabe o comentário de que o campo de aprendizado por reforço multi-objetivo ainda se encontra em seus primórdios [54], e as contribuições encontradas na literatura são muito fragmentadas. De forma geral, o desempenho dos algoritmos propostos é analisado apenas para um número reduzido de problemas e de forma isolada. Os resultados apresentam, portanto, duas limitações: o número de problemas explorados é bastante reduzido, o que impede a análise do comportamento dos algoritmos em uma gama maior de problemas, cuja estrutura é diferente daquele para o qual a solução foi proposta (em outras palavras, ainda não há um conjunto de problemas canônicos que devem ser resolvidos para que uma solução seja validada). Em segundo lugar, não é possível definir uma sobreposição clara entre os tipos de problema abordados e as metodologias propostas por cada um dos autores, o que torna a comparação das soluções impossível. Além disso, não há consenso entre quais são as métricas mais apropriadas para a comparação dos algoritmos propostos e a qualidade das soluções encontradas [54].

2.9.1 Algoritmos de Política Única

Nos algoritmos de política única, o agente aprende apenas uma política, de forma a satisfazer um conjunto de preferências entre os objetivos que já foi estabelecido *a priori* pelo usuário ou foi derivado da própria natureza do problema. A maior parte dos algoritmos propostos até o momento é dessa natureza [55, 56, 57].

A abordagem consiste em fazer com que um agente maximize uma função dos diferentes objetivos ou, de forma alternativa, o sinal recompensa que é percebido pelo agente é dado por $r(s,a) = F\left(\vec{r}(s,a)\right)$, em que F é uma função que mapeia o vetor $\vec{r}(s,a)$ em um escalar. A função mais simples a ser utilizada é uma combinação linear das recompensas nas diferentes dimensões do problema. Dado o vetor de recompensas $\vec{r}(s,a)$, faz-se com que a recompensa recebida pelo agente seja expressa por:

$$\begin{aligned} r_\alpha(s,a) &= \vec{\alpha}\cdot\vec{r}(s,a) \\ &= \alpha_1 r_1(s,a) + \alpha_2 r_2(s,a) + \cdots + \alpha_N r_N(s,a) \\ &= \sum_{i=1}^{N} \alpha_i r_i(s,a) \end{aligned} \tag{2.50}$$

Cabe notar que, para cada vetor de pesos $\vec{\alpha}$, existe uma política ótima π^*_α. Neste aspecto, essa classe de soluções lembra a abordagem mono-objetivo, e os algoritmos apresentados anteriormente podem ser utilizados para a solução dos problemas. É papel do usuário especificar os pesos $\vec{\alpha}$, de forma a expressar a importância relativa de cada objetivo. Consequentemente, pequenas mudanças no vetor de pesos pode produzir grandes mudanças na política aprendida, de

forma que uma mudança nas preferências requer que o processo de aprendizado seja reiniciado.

Ainda que seja considerada uma possível abordagem para o problema, os algoritmos de política única destacam-se quando os objetivos são diretamente relacionados, e podem, portanto, ser combinados em apenas um objetivo. Dessa forma, a política ótima encontrada maximizará a todos de forma simultânea.

2.9.2 Dominância de Pareto

As situações de interesse prático são aquelas em que os múltiplos objetivos a serem atendidos são conflitantes. Se todos os objetivos do problema estão relacionados e possuem dependência direta, de forma que a maximização de um dos objetivos implica também na maximização dos demais, então tem-se um caso de otimização mono-objetivo. Por outro lado, se os diferentes objetivos são conflitantes, então uma política deve ou maximizar apenas um objetivo, ou representar uma solução de compromisso entre as diferentes dimensões do problema. Uma forma tradicional de comparar soluções conflitantes entre os diferentes objetivos é por meio do conceito de dominância de Pareto.

Um ponto $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \end{bmatrix} \in \mathbb{R}^N$ domina, no sentido de Pareto, um outro ponto $\mathbf{x}' = \begin{bmatrix} x'_1 & x'_2 & \cdots & x'_N \end{bmatrix} \in \mathbb{R}^N$ se, e somente se, $\forall i x_i \geq x'_i$ e $\exists i x_i > x'_i$, para $1 \leq i \leq N$.

Este conceito está ilustrado na Fig. 2.2, em problema para o qual tem-se $N = 2$. Uma solução (*A*) domina fortemente outra solução (*C*) se é superior a esta em todos os objetivos. Uma solução (*B*) domina fracamente outra solução (*C*) se é superior em pelo menos um objetivo, mas possui o mesmo desempenho nos demais objetivos. Finalmente, duas soluções (*A* e *B*) são incomparáveis (ou equivalentes, ou ainda não dominadas) se não se dominam forte ou fracamante.

Uma solução dominada é de pouco valor pois, neste caso, é sempre preferível a escolha da solução que a domina. Logo, as melhores soluções podem ser obtidas a partir do conjunto que contém apenas as soluções que não se dominam. Este conjunto é denominado frente de Pareto e está ilustrado na Fig. 2.3. Do conjunto de soluções possíveis, aquelas representadas por pontos em preto pertencem à frente de Pareto, pois não se dominam. Estas mesmas soluções dominam forte ou fracamente as soluções representadas pelos pontos em cinza.

Estabelecer a frente de Pareto para problemas de grande dimensionalidade é uma tarefa impraticável, de forma que o objetivo de um algoritmo de otimização multi-objetico deve ser produzir um conjunto de soluções que se aproxima da frente de Pareto.

2.9.3 Algoritmos de Múltiplas Políticas

A extensão dos algoritmos de aprendizado por reforço do caso mono-objetivo para o caso multi-objetivo introduz outras possibilidades de variação dos algoritmos tradicionais. Nos

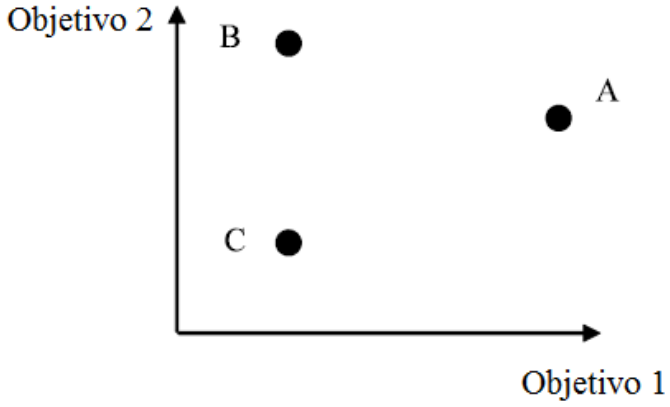


Figura 2.2: Ilustração do conceito de dominância entre soluções para um problema bidimensional. A solução A domina fortemente a solução C, a solução B domina fracamente a solução C, e as soluções A e B não se dominam, ou são incomparáveis.

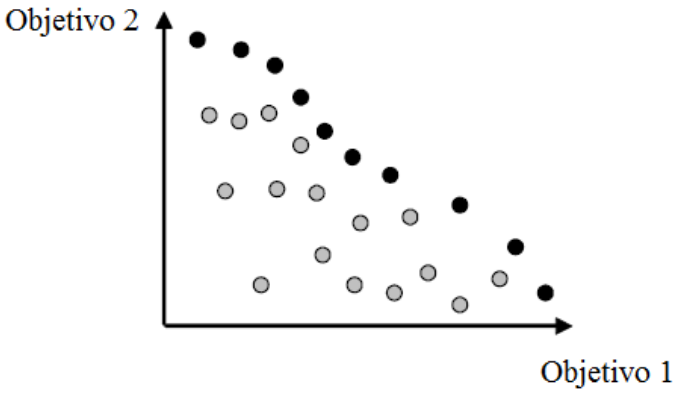


Figura 2.3: Ilustração da frente de Pareto para um problema bi-dimensional.

algoritmos mono-objetivos apresentados, todos buscam encontrar uma política que maximize uma soma acumulada de sinais de recompensa. Para o caso de problemas multi-objetivo, não existe apenas uma política ótima, pois um conjunto de políticas pode ser não dominada no sentido de Pareto. Neste caso, cabem aos novos algoritmos a descoberta dessas políticas.

Nos algoritmos de múltiplas políticas, o agente busca encontrar um conjunto de políticas não dominadas. Será apresentado o algoritmo de iteração no fecho convexo, apresentado em [53]. Nesta bordagem, o fecho convexo é utilizado para determinar quais políticas pertencem à frente de Pareto, aprendendo em paralelo um conjunto de políticas determinísticas.

Para o caso em que há múltiplas recompensas, pode-se escrever a função valor do estado como

$$\vec{V}^{\pi}(s) = \mathbb{E}\left\{\sum_{t=0}^{\infty} \gamma^t \vec{r}_t \,|s_{t=0} = s\right\} \tag{2.51}$$

de tal forma que a função valor-ação é

$$\vec{Q}^{\pi}(s,a) = \mathbb{E}\left\{\vec{r}_t(s,a) + \gamma\vec{V}^{\pi}(s') \,|s_{t=0} = s, a_{t=0} = a\right\} \tag{2.52}$$

Dado um conjunto de pesos $\vec{\alpha}$, $r(s,a) = \vec{\alpha}\cdot\vec{r}(s,a)$, e obtém-se, a partir da Eq. (2.52) e aplicando o princípio da otimalidade de Bellman, a forma

$$Q_{\vec{\alpha}}(s,a) = \mathbb{E}\left\{\vec{\alpha}\cdot\vec{r}(s,a) + \gamma\max_{a'} Q_{\vec{\alpha}}(s',a') \,|s_{t=0} = s, a_{t=0} = a\right\} \tag{2.53}$$

Para cada vetor de pesos $\vec{\alpha}$, existe uma política ótima π^*_{α}. O algoritmo de iteração no fecho convexo possui como objetivo não determinar qual a melhor ação para um vetor de pesos $\vec{\alpha}$ fixo, mas sim quais são as melhores ações independentemente da escolha de $\vec{\alpha}$.

O fecho convexo de um conjunto de M pontos $\mathbf{x}_1$, $\mathbf{x}_2$, $\cdots$, $\mathbf{x}_M$ em $\mathbb{R}^N$, representado por hull $\bigcup_{i=1}^{M}\mathbf{x}_i$, é definido como a menor região convexa em $\mathbb{R}^N$ que contém todos os pontos $\mathbf{x}_i$. Matematicamente, qualquer ponto do fecho convexo satisfaz [58]

$$\text{hull}\bigcup_{i=1}^{M}\mathbf{x}_i = \left\{\sum_{i=0}^{M}\beta_i\mathbf{x}_i \,\middle|\, \sum_{i=0}^{M}\beta_i = 1, \beta_i \geq 0\right\} \tag{2.54}$$

Em [53], mostra-se que, sob a restrição $\sum_{i=0}^{N}\alpha_i = 1$, os vetores $\vec{Q}(s,a)$ que são máximos para algum $\vec{\alpha}$ encontram-se sob a fronteira do fecho convexo de todos os valores $Q(s,a)$. Os pontos que se encontram sobre a fronteira do fecho convexo são máximos em alguma dimensão, que é um conceito similar à frente de Pareto, mas não se resume à mesma. Na verdade, o fecho convexo busca apenas aproximar a frente de Pareto utilizando procedimentos computacionalmente mais eficientes.

A iteração no fecho convexo é dada pela relação de recorrência definida como [53]

$$\overset{o}{Q}(s,a) \leftarrow \mathbb{E}\left\{\vec{r}(s,a) + \gamma\text{hull}\bigcup_{a'}\overset{o}{Q}(s',a') \,|(s,a)\right\} \tag{2.55}$$

em que $\overset{o}{Q}(s,a)$ representa os vértices do fecho convexo dos possíveis valores da função Q ao executar a ação a no estado s. Ainda em [53], mostra-se que a iteração dada pela Eq. (2.55) converge para a relação definida pela Eq. (2.53), mostrando a correção do algoritmo, consistindo em uma extensão do algoritmo da iteração de valor para aprender um conjunto de valores ótimos de Q para todos os conjuntos de $\vec{\alpha}$ possíveis.

Como comentário final, observa-se que, diferentemente do caso de algoritmos de política única, algoritmos de múltiplas políticas apresentam custo computacional mais elevado e devem interagir com o ambiente por um intervalo maior de tempo para o aprendizado das políticas

ótimas. Por outro lado, algoritmos de política única, ao trabalharem com preferências do usuário, possuem custo computacional reduzido e podem ser mais indicados para aprendizado *on-line* [54].

2.10 CONCLUSÃO

Este capítulo apresentou os principais conceitos relacionados ao aprendizado por reforço, seja em suas versões tabular, de estados contínuos ou multi-objetivo. Nos próximos capítulos, a modelagem aqui exposta será utilizada para o tratamento de problemas na área de sistemas de comunicação, a citar a modulação e codificação adaptativas em sistemas OFDM, a alocação de potência nesses mesmos sistemas, e o escalonamento de usuários em um ambiente de comunicação celular.

3 IMPLEMENTAÇÃO DA ESTRATÉGIA DE MODULAÇÃO E CODIFICAÇÃO ADAPTATIVAS POR MEIO DE APRENDIZADO POR REFORÇO

3.1 INTRODUÇÃO

Este capítulo apresenta a modelização, em sistemas OFDM, da técnica de adaptação de enlace denominada modulação e codificação adaptativas como um problema de aprendizado por reforço de estados contínuos.

O conceito fundamental da estratégia de modulação e codificação adaptativas (AMC, *adaptive modulation and coding*) é explorar a informação disponível sobre o estado do canal de comunicação (CSI, *channel state information*) para que os parâmetros de transmissão referentes à modulação e à codificação de canal utilizados pelos terminais de comunicação sejam modificados, de forma a maximizar a vazão do enlace de comunicação [59, 60, 61, 62, 63]. Em outras palavras, busca-se alterar os parâmetros de modulação e codificação de canal de acordo com as variações do canal de comunicação, de forma a obter um aumento na vazão do sistema de comunicação. A aplicação do AMC é de grande interesse prático, especialmente ao considerar os requisitos de vazão e taxa de transmissão para os sistemas de terceira e quarta geração (3G e 4G, respectivamente) [64].

Atualmente, a abordagem mais utilizada para a implantação da estratégia de AMC é por meio de tabelas de consulta [32]. Nessas tabelas, faz-se a relação entre qual a melhor modulação e codificação a serem utilizadas para um determinado estado do canal de comunicação, sendo este estado medido, em geral, em função da razão sinal-ruído (SNR, *signal-to-noise ratio*). Um dos problemas para a elaboração de tabelas de consulta, especialmente para sistemas OFDM codificados ou que utilizam múltiplas antenas (MIMO, *multiple-input multiple-output*), é a obtenção de métricas apropriadas, ou LQM (*link quality metrics*), para descrever a qualidade do enlace. Um segundo problema da abordagem por tabelas de consulta é a geração das mesmas. Elas não são elaboradas em tempo real, mas sim dependem de um conjunto muito grande de simulações do sistema de comunicação em questão sob diferentes condições de operação (em termos de tamanho dos pacotes utilizados, resposta do canal, algoritmos de detecção etc.) [65]. Como consequência, as tabelas de consulta demandam grande quantidade de memória para seu armazenamento e podem não refletir características específicas dos dispositivos de comunicação [66]. Em situações práticas, os limiares de razão sinal-ruído são determinados de forma heurística utilizando dados coletados a longo prazo nas interfaces de rádio [67]. Esta estratégia requer certo

grau de perícia do operador, e não garante a maximização da vazão do sistema, uma vez que um número muito grande de situações é utilizado para obter valores razoáveis para serem utilizados como limiares. Consequentemente, os valores obtidos podem levar a uma seleção muito otimista ou muito pessimista dos esquemas de modulação e codificação para um dado valor de razão sinal-ruído.

A partir do exposto, percebe-se que as formas atualmente utilizadas de implementação de modulação e codificação adaptativas estão baseadas em soluções *off-line* e utilizam-se de modelos idealizados dos terminais de comunicação e do canal de comunicação. Sistemas reais e transceptores práticos estão sujeitos a efeitos não lineares que não são capturados por estes modelos, além do fato de que o próprio sistema de comunicação evolui e varia com o tempo, seja em termos de variação do canal, seja em termos de níveis de interferência cuja distribuição estatística não é branca e gaussiana. Uma das formas de superar essa deficiência é por meio de técnicas de aprendizado de máquina e inteligência artificial.

Este capítulo está organizado da seguinte forma: inicialmente, faz-se uma revisão bibliográfica das formas propostas de realização de adaptação de enlace por meio da modulação e codificação adaptativas propostas que se utilizam de inteligência artificial. Em seguida, é apresentada a camada física do sistema de comunicação para o qual a modulação e codificação adaptativas são propostas. Posteriormente, é apresentada a primeira contribuição do capítulo, que consiste na formalização da solução do problema de AMC utilizando uma abordagem por aprendizado por reforço, identificando as ações, estados e recompensas que caracterizam o processo de decisão Markoviano utilizado. É então feita a segunda contribuição do trabalho, que consiste na proposta de modificação do algoritmo LSPI, exposto no Capítulo 2, de forma a adaptá-lo ao caso em que é necessário o aprendizado *on-line*. Finalmente, as propostas são avaliadas por meio de simulação em diferentes cenários de comunicação, considerando a comparação de seu desempenho com a solução mais tradicional adotada atualmente, que tem como base a utilização de tabelas de consulta.

3.2 REVISÃO BIBLIOGRÁFICA

Como anteriormente citado, a abordagem mais utilizada para a implantação da estratégia de AMC é por meio de tabelas de consulta. Estas se utilizam de modelos teóricos idealizados ou resultados de simulação como forma de determinar quais as melhores técnicas de modulação e codificação a serem utilizadas de acordo com o comportamento do canal. Entretanto as tabelas de consulta consistem em uma solução subótima se forem aplicadas em ambientes cujas características são diferentes das hipóteses dos modelos utilizados para sua obtenção.

Em [68], é proposta uma mudança no paradigma utilizado. Os autores sugerem que técnicas de aprendizado de máquina podem ser utilizadas como um possível *framework* para a solução

do problema de modulação e codificação adaptativas, possuindo a flexibilidade que não existe na solução por tabelas de consulta. Em [68, 66], o AMC é formulado como um problema de classificação, cuja solução é obtida pela utilização do algoritmo *kNN* (*k-nearest neighbors*). Em [69], máquinas de vetor de suporte são utilizadas para resolver o mesmo problema de classificação proposto, enquanto que em [70] é utilizada uma abordagem por redes neurais para lidar com o problema de adaptação de enlace.

A aplicação de algoritmos de aprendizado de máquina como os citados anteriormente, assim como os demais algoritmos de aprendizado supervisionado, depende fortemente do dados do conjunto de treinamento utilizados para o aprendizado e requer um grande número de amostras do par entrada-saída das funções a serem aprendidas. Logo, informações como a taxa de erro de pacotes (PER, *packet error rate*) ou a taxa de erro de *bit* (BER, *bit error rate*) em função das métricas de qualidade do enlace devem ser conhecidas *a priori*. Além disso, as soluções apresentadas dependem de uma fase de aprendizado que é relizada *off-line*, o que não as torna muito apropriadas para o aprendizado em ambientes muito dinâmicos ou de grande variabilidade, como é o caso do canal de comunicação móvel sem fio. O processo de treinamento as redes neurais e das máquinas de vetor pode ser computacionalmente intensivo, assim como ocorre com as tabelas de consulta [71].

Cabe ainda mencionar que, frequentemente, não é simples obter um conjunto de treinamento que seja apropriado, de forma a representar um número grande de cenários no qual transmissor e receptor se encontram. Como exemplo, pode-se citar o comportamento do canal de comunicação sem fio, o impacto de amplificadores não lineares, variação no ruído de fase do sistema receptor e imperfeições de RF em geral [72], bem como a presença de ruído não gaussiano e de interferência de outros sistemas. Este, em particular, é de especial interesse nos cenários de comunicação por rádio cognitivo, nos quais o comportamento da interferência pode divergir da modelagem tradicionalmente utilizada devido à própria natureza adaptativa dos rádios cognitivos [73]. Nesta situação, a hipótese de que a interferência pode ser modelada como um processo gaussiano nem sempre é válida [74].

O fato de que as soluções citadas não são viáveis para o aprendizado *on-line* e dependem fortemente de um conjunto de treinamento sugere que outras abordagens podem ser utilizadas para a solução do problema de AMC. Neste contexto, será apresentado como primeira contribuição do trabalho uma implementação da estratégia de modulação e codificação adaptativas utilizando uma abordagem por aprendizado por reforço. No melhor do conhecimento do autor, o tratamento do problema de AMC por meio da utilização de aprendizado por reforço por estados contínuos ainda não foi explorada na literatura. Como segunda contribuição, é apresentada uma modificação do algoritmo LSPI para que este possa ser utilizado de forma *on-line* para a solução do problema de aprendizado.

Desse forma, com a abordagem por aprendizado por reforço, o agente, utilizando a experiência aprendida de forma direta, por interações com o ambiente, é capaz de aprender qual

o melhor esquema de modulação e codificação para cada estado de canal, utilizando o menor número possível de hipóteses acerca do ambiente de rádio. Neste caso, o objetivo do agente é maximizar a eficiência espectral do sistema.

3.3 MODELAGEM DO PROBLEMA E SOLUÇÃO PROPOSTA

Nesta seção, a adaptação de enlace via modulação e codificação adaptativas é modelada como um problema de aprendizado por reforço. Diferentemente das soluções apresentadas na literatura que também se utilizam de técnicas de aprendizado de máquina e foram expostas brevemente na Seção 3.2, a solução proposta:

- utiliza aprendizado por reforço e, como não é uma técnica supervisionada, não depende da existência de conjuntos de treinamento para realizar o aprendizado;
- faz uso de uma abordagem que utiliza valores contínuos para descrever os estados do sistema;
- consiste em uma versão modificada do algoritmo LSPI para ser capaz de promover o aprendizado em tempo real;
- apresenta potencial capacidade de adaptação a diferentes tipos de ambiente de rádio e características específicas de transceptores, o que é uma característica desejável em ambientes dinâmicos ou em ambientes cuja modelagem analítica pode ser muito complexa, como é o caso da presença de interferência colorida.

3.3.1 Camada Física de Transmissão

Será considerado um sistema de comunicação OFDM cuja camada física é semelhante à do padrão LTE (*Long Term Evolution*), proposto pelo 3GPP (*Third-Generation Partnership Project*), cujo detalhamento se encontra no Anexo I. A transmissão dos *bits* de informação é feita por meio de pacotes de dados. A cada pacote, é adicionado um campo de CRC (*cyclic redundancy check*) e, em seguida, toda essa informação é apresentada à entrada de um codificador de canal do tipo convolucional. Os *bits* codificados são usados para modular uma portadora e o sinal resultante é utilizado para a formatação dos símbolos OFDM, que serão acomodados em *resource blocks*.

A modulação é uniforme no sentido que todas as subportadoras de um mesmo *resource block* são moduladas pela mesma constelação M-QAM (*M-ary quadrature amplitude modulation*). Supõe-se ainda que, ao formatar o símbolo OFDM, é inserido um intervalo de guarda apropriado de forma a compensar a interferência inter-simbólica introduzida pelos múltiplos percursos do canal de comunicação sem fio.

Um diagrama de blocos que ilustra a operação do sistema é mostrado na Fig. 3.1. É evidenciada a presença de um agente inteligente que controla a escolha das combinações de modulação e codificação a serem utilizadas e, para tal, depende da presença de um canal de retorno no sistema e deve ser capaz de estimar a resposta do canal multipercurso a fim de calcular a razão sinal-ruído média em cada *resource block*. Os detalhes específicos do procedimento serão dados na Seção 3.3.2 e também na Seção 3.4, onde é abordado o cenário completo de simulação.

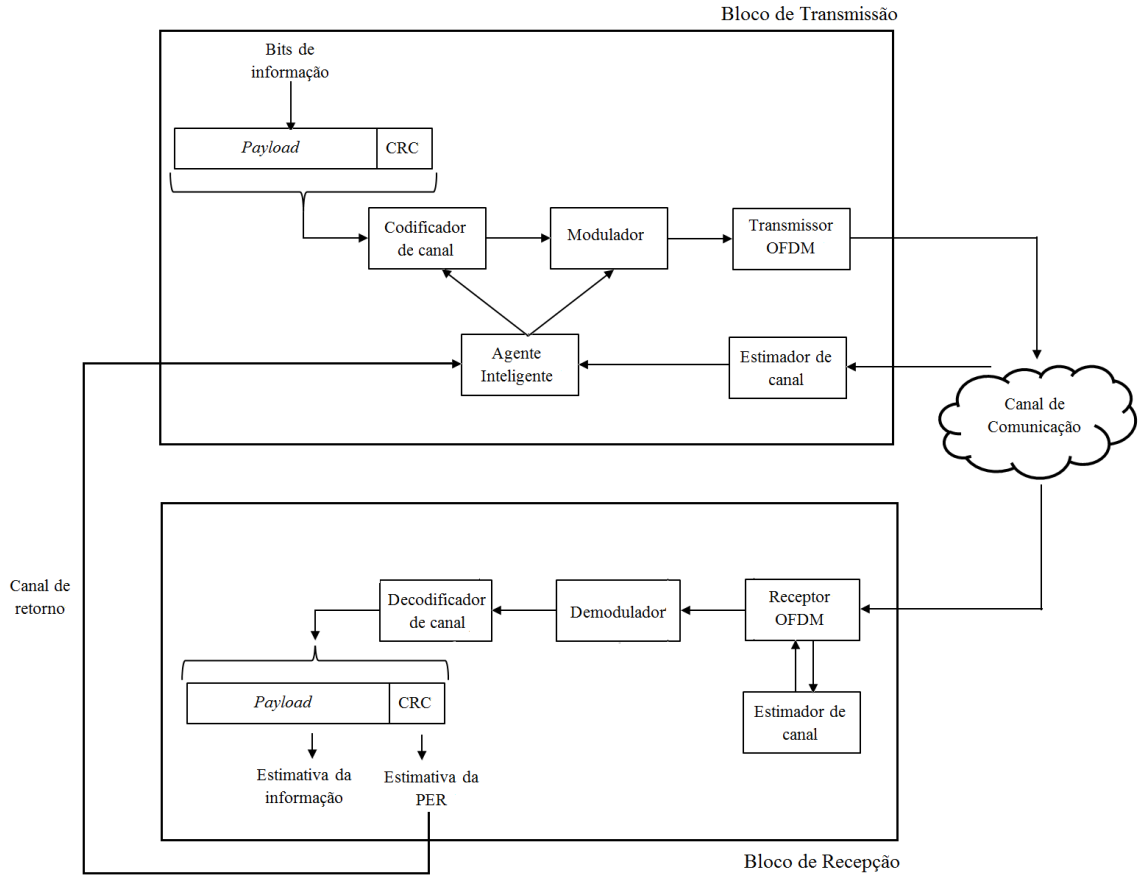


Figura 3.1: Diagrama de blocos do sistema de transmissão OFDM considerado para análise.

Quanto ao comportamento do canal de comunicação, supõe-se que sua resposta em frequência pode variar consideravelmente entre símbolos OFDM distintos (dependendo da velocidade relativa entre os terminais de transmissão e de recepção), mas não varia dentro do horizonte de um símbolo OFDM (isto é, utiliza-se a hipótese de desvanecimento por blocos quase-estático) [75].

No receptor, o bloco denominado *Receptor OFDM* é responsável por identicar a formatação do quadro de transmissão e equalizar o sinal recebido utilizando um equalizador de canal do tipo *zero-forcing* (motivo pelo qual a estimação da resposta do canal é também necessária). Em seguida, o sinal resultante é detectado pela regra MAP (*maximum a posteriori*) no demodulador,

e o decodificador de canal utiliza o algoritmo de Viterbi para recuperar os *bits* de informação.

3.3.2 Ações, Estados e Recompensas

Diferentemente das abordagens mais tradicionais de solução via aprendizado por reforço, baseadas em um conjunto de estados discretos, optou-se por utilizar uma abordagem que utiliza um conjunto contínuo de estados. A escolha justifica-se pois, ao utilizar um número discreto de estados, pode ser difícil determinar a melhor partição do espaço de estados. Como os limiares de razão sinal-ruído utilizados para a transição entre as diferentes combinações de modulação e codificação não são conhecidos *a priori*, uma discretização grosseira pode levar à perda de capacidade em uma região específica de operação. Uma discretização muito granular elava consideravelmente o custo computacional, o espaço de armazenamento das políticas e o tempo de convergência do algoritmo.

Para o enlace ponto a ponto apresentado, o objetivo é maximizar a vazão do sistema para um dado estado do ambiente, o que caracteriza um problema do tipo N*-armed bandit* [47], no qual as ações a_t não influenciam o estado futuro s_{t+1} do ambiente. Nesta situação, o ambiente é representado pelo canal de comunicação e seu descritor de estados é a razão sinal-ruído média tomada sobre todas as subportadoras presentes em um *resource block*. Ainda que seja uma métrica de qualidade de enlace muito simples, ela foi escolhida por ser uma solução de compromisso entre seu custo computacional e qualidade no ajuste das curvas de taxa de erro de pacote [32].

Não é objetivo deste trabalho verificar quais as melhores métricas de descrição da qualidade de enlace, ainda que a escolha de uma métrica apropriada pode ser crítica para o caso de sistemas de comunicação que utilizam OFDM e adaptação de enlace, como apresentado em [68]. Em vez disso, busca-se sobretudo uma solução para a estratégia de AMC que não dependa de um grande conjunto de dados de treinamento, realizado de forma *off-line*, e que seja capaz de se adaptar a diferentes condições do canal de comunicação.

As ações a serem realizadas pelo agente pertencem ao conjunto $\mathcal{A}$ que consiste nas combinações possíveis entre as diversas modulações e taxas de codificação que podem ser utilizadas pelo sistema, e serão explicitadas na Seção 3.4.

A função recompensa proposta por essa tese é definida como o *goodput* alcançado ao executar a ação a quando o ambiente se encontra no estado s, e é dada por:

$$R\left(s,a,s'\right) = \log_2\left(M_a\right)\rho_a\left[1 - PER\left(s,a\right)\right] \tag{3.1}$$

em que M_a é a ordem da modulação da ação a, ρ_a é a taxa de codificação da ação a, e $PER\left(s,a\right)$ é a taxa de erro de pacote resultante de tomar a ação a quando em s. Uma vez que há um campo CRC em cada pacote transmitido, o receptor pode identificar os pacotes que foram recebidos com erro [76] e a PER pode ser estimada diretamente a partir de observações a longo prazo, conforme

mostrado em [77] e [78], e a estimativa pode então ser enviada para o transmissor por meio de um canal de retorno.

3.3.3 Funções de Base

Conforme exposto no início da Seção 3.3, a solução proposta está baseada no aprendizado por reforço de estados contínuos. De acordo com a teoria apresentada no Capítulo 2, a formalização do problema requer, além da descrição dos estados, ações e recompensas (esta realizada na Seção 3.3.2), as funções de base que serão utilizadas pela arquitetura linear de aproximação de funções. Em outras palavras, busca-se a aproximação de $Q(s,a)$ por meio de $Q^{\pi}\left(s,a\,;\mathbf{w}\right)$ dado por:

$$Q^{\pi}\left(s,a\,;\mathbf{w}\right)=\begin{bmatrix}\phi_1\left(s,a\right) & \cdots & \phi_k\left(s,a\right)\end{bmatrix}\cdot\begin{bmatrix}w_1\\ w_2\\ \vdots\\ w_k\end{bmatrix} \tag{3.2}$$

em que $\phi_l(s,a)$, com $l=1,\cdots,k$, são as funções de base.

Um dos problemas comuns encontrados em inteligência artificial é o de aproximação de funções para problemas mal formulados [1], cuja solução é formalizada pela teoria da regularização [79]. Uma classe particularmente interessante de funções $\phi_l(s,a)$ que podem ser utilizadas são as funções de base radial, por serem invariantes à rotação e à translação e fornecerem boa capacidade de generalização local (isto é, mudanças em uma determinada região do espaço de estados não afetam todo o espaço de estados [47]).

Uma das classes de funções de base radial é a das funções gaussianas, dadas por:

$$\phi_l(s,a)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(s-\mu_l)^2}{2\sigma^2}} \tag{3.3}$$

em que μ_l é o parâmetro de localização e σ é o parâmetro de forma, que reflete o campo perceptivo (ou campo de influência) da função base [79].

Apenas a título de ilustração, será considerado o exemplo hipotético mostrado na Fig. 3.2, que mostra o conceito de aproximação de uma função a partir da expansão mostrada na Eq. (3.2). A Fig. 3.2(a) mostra as amostras disponíveis da função a ser aproximada no intervalo $-3\leq s\leq 3$. Para realizar tal aproximação, foram utilizadas três funções de base gaussianas, mostradas na Fig. 3.2(b). Todas as três funções possuem parâmetro $\sigma^2=1$ e parâmetros de localização dados por $\mu_1=-1,5$;$\mu_2=0$ e $\mu_3=2$. Na Fig. 3.2(c) é mostrado o resultado na aproximação dado pela

[1] O termo mal formulado é empregado no sentido de que as amostras que estão disponíveis para a aproximação da função em questão pode não conter toda a informação necessária para a correta aproximação ou interpolação de valores. O termo ainda se refere aos casos em que as amostras podem estar contaminadas por ruído (o que é fato em qualquer cenário prático), de forma que há imprecisão inerente aos pares de entrada e saída que são utilizados para aproximação uma dada função.

Eq. (3.2) utilizando-se os pesos $w_1 = 1$, $w_2 = 1$ e $w_3 = 0,5$. A Fig. 3.2(d) procura ilustrar a influência de cada base na forma final da função aproximada.

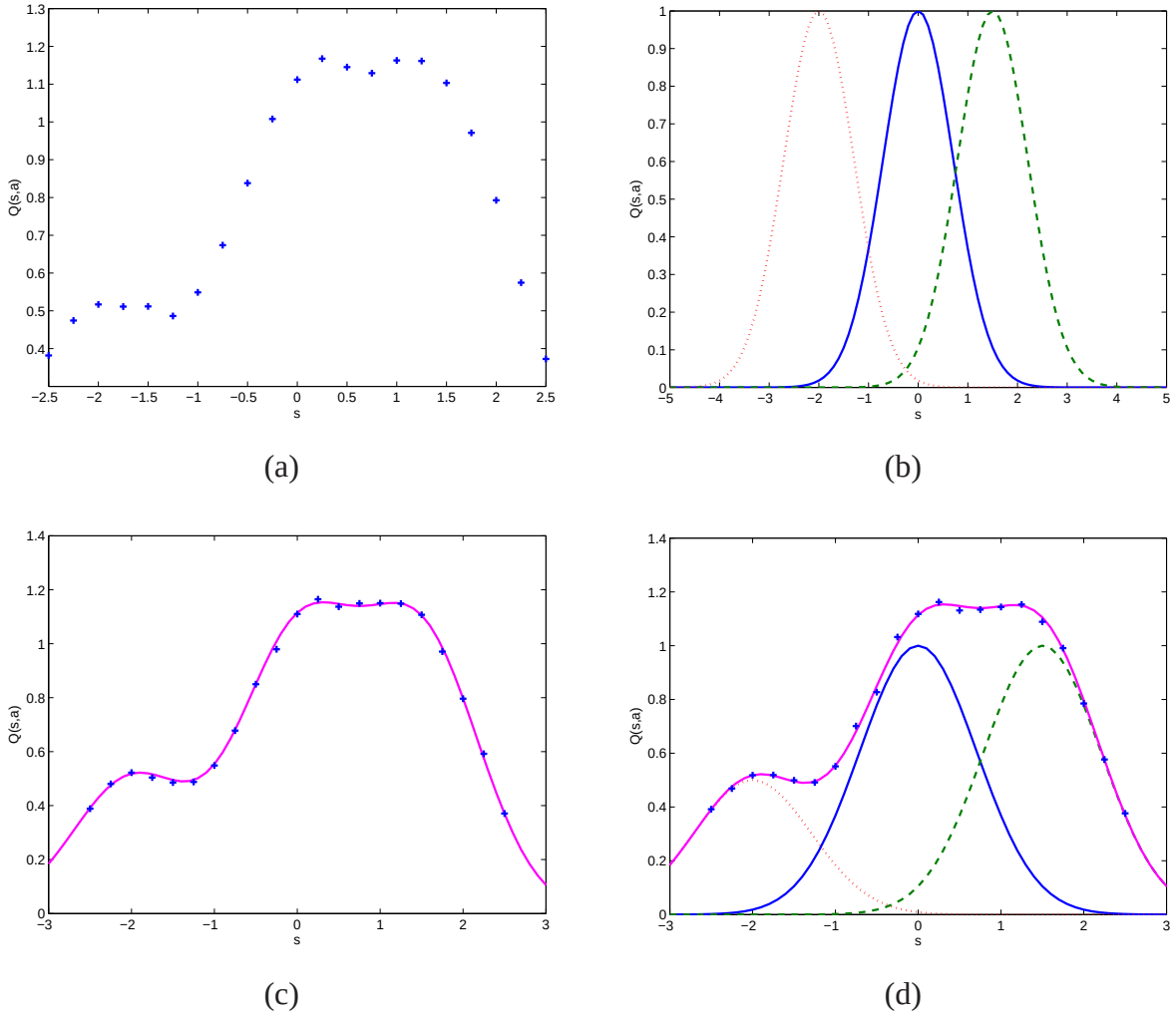


Figura 3.2: Exemplo (hipotético) do problema de aproximação de funções. Em (a), é mostrado o conjunto de dados disponíveis da função a ser aproximada. Em (b), tem-se o conjunto de bases utilizado na aproximação. Finalmente, (c) e (d) mostram o resultado da aproximação e a influência de cada base no resultado final.

Para o problema de aprendizado por reforço em questão, foram escolhidas $k = 5$ funções de base para cada uma das ações disponíveis ao agente. Este valor foi determinado a partir de vários testes conduzidos, considerando uma solução de compromisso entre a precisão dos resultados obtidos e a complexidade computacional resultante na execução do algoritmo (esta a ser abordada na Seção 3.3.5).

A base da arquitetura linear de aproximação, conforma definida pela Eq. (2.25), é dada por cinco funções de base radial. Para uma ação a,

$$\boldsymbol{\phi}(s,a) = \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(s-\mu_1)^2}{2\sigma^2}} \quad \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(s-\mu_2)^2}{2\sigma^2}} \quad \dots \quad \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(s-\mu_5)^2}{2\sigma^2}} \right]^T \tag{3.4}$$

em que os parâmetros de localização $\mu_1, \dots, \mu_5$ estão igualmente espaçados na região definida pelos valores $0\text{dB} \leq SNR \leq 40\text{dB}$, que foi o intervalo prático de valores de razão sinal-ruído considerados para a elaboração da solução. O parâmetro de escala foi escolhido de forma empírica como $\sigma^2 = 8$ [45], de forma a não permitir influência excessiva entre as bases.

3.3.4 Proposta de Modificação do Algoritmo LSPI

Apesar de o algoritmo LSPI, conforme mostrado no Capítulo 2, ser considerado o estado da arte no que diz respeito aos algoritmos de melhora de política para problemas de aprendizado por reforço de estados contínuos [52], ele apresenta como desvantagem o fato de não trabalhar de forma *on-line*. A melhoria de política só é realizada após repetidas execuções do algoritmo LSTD-Q em um *batch* de amostras do ambiente para que uma aproximação acurada da função Q possa ser obtida.

Por outro lado, um dos atrativos da modelagem por aprendizado por reforço é aprender a partir das iterações com o ambiente e buscar por políticas ótimas de maneira *on-line* (preferencialmente) [80], ainda que existam aplicações que requeiram o processamento das medidas do ambiente de forma *off-line*. Tendo como justificativa esta preferência pelo aprendizado *on-line*, esta seção propõe uma versão modificada do algoritmo LSPI, que inclui uma estratégia de exploração ϵ-*greedy* para a melhoria de políticas [81]. Dessa forma, o aprendizado pode se dar em tempo real.

O detalhamento da versão modificada do LSPI para lidar com problemas do tipo N-*armed bandit* é mostrado no Algoritmo 3. Apesar de suas semelhanças com o LSPI, há duas diferenças importantes: o passo 5, que não está presente no algoritmo original, implementa uma estratégia de exploração ϵ-*greedy* para resolver o dilema da exploração versus explotação. O passo 10 realiza a busca pela política gulosa.

Em geral, é requerida grande quantidade de exploração para o conhecimento do comportamento do ambiente, mas a exploração não pode permanecer alta por toda a execução do algoritmo, caso contrário a política ótima não será propriamente explotada. Considera-se ainda, para o caso específico do problema de modulação e codificação adaptativas, que as características do canal de comunicação podem variar com o tempo. Dessa forma, é interessante manter algum grau de exploração para que o algoritmo possa rastrear possíveis mudanças no comportamento do ambiente.

Buscando uma solução de compromisso entre esses dois fatores, sugere-se que a probabilidade ϵ de escolha de uma ação aleatória, conforme exposto pela Eq. (2.17), seja uma função da iteração

t dada por:

$$\epsilon_t = \max\left(\epsilon_f, \epsilon_i^{-\tau t}\right), \tag{3.5}$$

em que $\epsilon_f < \epsilon_i \in [0;1]$, ϵ_i representa o valor inicial de probabilidade de exploração e τ é uma constante que representa o fator de decaimento de exploração. A essência da exploração dada pela Eq. (3.5) é: durante as primeiras iterações do algoritmo, os valores de ϵ_t são próximos dos valores de ϵ_i. Se ϵ_i possui valor próximo de um, o agente possui estratégia de exploração aleatória mais agressiva. Com o passar das iterações, ϵ_t decai, de tal forma que a exploração diminui. O valor de ϵ_t continua decaindo até que seja igual a ϵ_f (que representa o limite inferior para a probabilidade de exploração). Como o valor de ϵ_f é próximo de zero, a explotação do ambiente passa a ser predominante. De acordo ainda com a Eq. (3.5), escolhendo $\epsilon_f \neq 0$ sempre garante uma certa probabilidade ϵ_f de exploração, mesmo após a convergência dos valores Q.

Algoritmo 3 *LSPI Modificado*

1. A política atual π é inicializada de forma aleatória
2. A matriz **A** e o vetor **b** são inicializados com
   $\mathbf{A}_{-1} = \delta \mathbf{I}_{k\times k}$
   $\mathbf{b}_{-1} = \mathbf{0}_{k\times 1}$
   em que δ é uma constante da ordem de 10^{-6}, $\mathbf{I}_{k\times k}$ é a matrix identidade $k \times k$ e $\mathbf{0}_{k\times 1}$ é um vetor $k \times 1$ de elementos nulos.
3. Para $t \geq 0$:
   4. O agente percebe o estado atual do ambiente, s_t
   5. Uma ação aleatória a_t é realizada com probabilidade ϵ_t, e a ação gulosa é realizada com probabilidade $1 - \epsilon_t$.
   6. Como resultado da ação, é gerado um sinal de recompensa r_t
   7. $\mathbf{A}_t = \mathbf{A}_{t-1} + \boldsymbol{\phi}\left(s_t, a_t\right)\boldsymbol{\phi}^T\left(s_t, a_t\right) - \gamma\boldsymbol{\phi}\left(s_t, a_t\right)\boldsymbol{\phi}^T\left(s_t, \pi\left(s_t\right)\right)$
   8. $\mathbf{b}_t = \mathbf{b}_{t-1} + \boldsymbol{\phi}\left(s_t, a_t\right) r_t$
   9. $\hat{\mathbf{w}} = \mathbf{A}_t^{-1}\mathbf{b}_t$
   10. Melhore a política atual utilizando $\pi\left(s\right) = \max_{a\in\mathcal{A}} \boldsymbol{\phi}^T\left(s, a\right)\hat{\mathbf{w}}$
11. Fim

3.3.5 Complexidade Computacional

Será descrita brevemente a complexidade computacional do sistema de aprendizado por reforço proposto, considerando o número de multiplicações como a métrica de complexidade. De acordo com o Algoritmo 3, a parte mais onerosa do algoritmo é a resolução do sistema linear resultante. Dependendo da técnica utilizada, a complexidade pode variar entre $\mathcal{O}\left(k^2\right)$ e $\mathcal{O}\left(k^3\right)$, enquanto que o produto interno entre as funções de base do sistema é de complexidade de ordem $\mathcal{O}\left(k\right)$. Dessa forma, o custo computacional está relacionado com o número de funções de base escolhidas na aproximação por arquitetura linear.

Tomando $k = 5$, que é a solução apresentada, é possível, sem gerar computações onerosas para uma estação rádio base, a implementação do esquema de AMC proposto, estando restrita apenas à memória disponível, visto que é necessário armazenar um vetor de pesos $\hat{\mathbf{w}}$ para cada usuário presente na célula, pois é este vetor de pesos que define qual a política (ótima) a ser adotada como forma de maximizar a eficiência espectral da transmissão.

3.4 AVALIAÇÃO DA PROPOSTA

Nesta seção, a proposta de modulação e codificação adaptativas utilizando o algoritmo proposto de aprendizado por reforço de estados contínuos é simulada, e seu desempenho é comparado com a abordagem tradicional de tabela de consulta.

3.4.1 Parâmetros e Cenário de Simulação

Para propósitos de simulação, definiu-se a forma de transmissão semelhante à camada física do padrão 3GPP-LTE para o enlace direto. O sistema ocupa uma largura de banda de 10 MHz e trabalha em FDD (*frequency-division duplexing*), no qual um quadro de trasmissão possui 10ms e é subdividido em 10 subquadros de 1ms cada. Cada subquadro é dividido em 2 *slots*, cada um contendo 6 símbolos OFDM.

O espaçamento entre as subportadoras foi fixado em 15 kHz, e a duração do prefixo cíclico foi de aproximadamente 4.6 μs. Os pacotes de informação são alocados em *resource blocks*, que são definidos como um conjunto de 7 símbolos OFDM no domínio temporal e 12 subportadoras no domínio da frequência, em um mesmo subquadro.

Tanto transmissor quanto receptor utilizam apenas uma antena. As combinações permitidas entre as diferentes formas de modulação de codificação são mostradas na Tabela 3.1, e são as ações que podem ser tomadas pelo agente. O código corretor de erros é do tipo convolucional de taxas 1/2, 2/3 ou 3/4. O codificador de canal base é de taxa 1/2 e polinômios geradores dados por $[133, 171]$ (em octal), e as taxas de 2/3 ou 3/4 são obtidas por meio da perfuração desse código base.

Para tornar as simulações mais realistas, utilizou-se um modelo de canal multipercursos variante no tempo denominado SCM (*Spatial Channel Model*), que gera os coeficientes de canal de acordo com o modelo padronizado pelo 3-GPP [82, 83]. Os parâmetros ajustados desse modelo de canal encontram-se na Tabela 3.2, e foram utilizados em todas as simulações, exceto quando especificado o contrário.

O conjunto de combinações possíveis entre técnicas de modulação e taxas de codificação é mostrado na Tabela 3.1, o que fornece $m = 6$ ações ou esquemas de AMC disponíveis ao agente.

Tabela 3.1: Esquemas de Modulação e Codificação

Número do esquema (Ação)	Modulação	Taxa de codificação
1	QPSK	1/2
2	QPSK	3/4
3	16QAM	1/2
4	16QAM	3/4
5	64QAM	2/3
6	64QAM	3/4

Tabela 3.2: Parâmetros do Canal SCM.

Parâmetro	Valor
Frequência de transmissão	2.0 GHz
Velocidade do terminal móvel	10.8 m/s
Número de antenas na estação base	1
Número de antenas na estação móvel	1
Cenário	macrocélula suburbana
Número de multipercursos	19

De forma a permitir a exploração de outras políticas, selecionou-se $\epsilon_f = 0,05$ (a probabilidade de exploração residual), $\epsilon_i = 0,95$ (a probabilidade inicial de exploração) e $\tau = 0,01$ (taxa de decaimento da probabilidade inicial de exploração), conforme definidos pela Eq. (3.5). Essas escolhas serão justificadas na análise dos resultados de simulação.

3.4.2 Tabela de Consulta

A técnica conhecida como mapeamento *RawBER* [32] foi utilizada para gerar a tabela de consulta para realizar a adaptação de enlace via modulação e codificação adaptativas. Neste tipo de mapeamento, a métrica de qualidade do enlace é encontrada por meio da probabilidade de erro de *bit* não codificado média para todo o conjunto de subportadoras de um *resource block* [65]. A relação entre a métrica de *RawBER* e a taxa de erro de pacote é realizada por meio de regressão entre a SNR média das subportadoras dentro de um *resource block* e os valores de PER obtidos a partir da simulação Monte Carlo de várias realizações do canal [84, 85], considerando ainda que na entrada do sistema receptor está presente, além do sinal transmitido, ruído branco aditivo gaussiano. Os limiares de razão sinal-ruído utilizados para a chaveamento entre os esquemas de modulação e codificação são determinados tomando como base um valor limite de taxa de erro de pacote de 10%. É necessário ainda, para cada simulação, fixar o tamanho do pacote de dados.

Uma das desvantagens da tabela de consulta é que o desempenho do sistema depende também do comportamento estatístico da interferência ao qual o mesmo está sujeito [50] e, por vezes, a hipótese de que o sinal interferente pode ser tratado como ruído brando nem sempre é válida

[74]. Devido ao grande esforço computacional que é necessário para gerar as tabelas mediante simulação, é impraticável gerar dados para cobrir todas as situações às quais o receptor pode estar sujeito.

3.4.3 Resultados de Simulação

A Fig. 3.3 mostra os resultados referentes à eficiência espectral média e à taxa de erro de pacote em função da razão sinal-ruído. Uma vez que a técnica de aprendizado por reforço foi aplicada sob as mesmas condições para as quais a tabela de consulta foi obtida e a mesma métrica de qualidade de enlace foi utilizada para realizar o mapeamento da razão sinal-ruído efetiva, o desempenho de ambas as técnicas é o mesmo. A principal diferença entre as abordagens é que o aprendizado por reforço opera em tempo real e não necessita de um supervisor (como ocorre nas técnicas de aprendizado supervisionado) ou de um conjunto extensivo de resultados de simulação sobre diferentes cenários (como é o caso das tabelas de consulta). O melhor esquema de modulação e codificação para um dado valor de razão sinal-ruído é determinado por meio de um procedimento de busca não exaustivo de diferentes ações, o que requer pouco esforço computacional.

A Fig. 3.4 e a Fig. 3.5 consideram os efeitos do ajuste do fator de desconto γ e da probabilidade de exploração ϵ da política ϵ-*greedy* na convergência do algoritmo de aprendizado por reforço. O erro quadrático médio (MSE, ou *mean ssquare error*) foi calculado considerando-se a diferença de vazão observada no sistema entre a política que era seguida quando da transmissão de um determinado quadro e o maior valor de vazão que poderia ser alcançado para um dado valor de razão sinal-ruído. Como resultado final, tomou-se a diferença média de vazão sobre todos os estados observados até o momento.

Como mostra a Fig. 3.4, quanto maior o fator de desconto, mais acelerada é a convergência do algoritmo, entretanto maior é o erro quadrático médio final após a convergência. Como esperado, um valor menor para o fator de desconto acarreta em um comportamento míope do algoritmo, uma vez que o agente passa a valorizar o valor das recompensas imediatas quando comparada às recompensas a longo prazo. De acordo com a Eq. (2.46) e a Eq. (2.47), apresentadas no Capítulo 2, quando menor o valor de γ, menores são os passos de atualização das matrizes $\widetilde{\mathbf{A}}$ e $\widetilde{\mathbf{b}}$, justificando a convergência mais lenta do algoritmo.

A Fig. 3.5 mostra o efeito do ajuste dos valores de ϵ_i e ϵ_f na convergência do algoritmo de aprendizado por reforço. Como ilustra a Fig. 3.5(a), é interessante que, inicialmente, o algoritmo apresente uma política de exploração mais agressiva. Procedendo dessa forma, o agente é capaz de aprender de forma mais rápida quais esquemas de modulação e codificação (ou seja, ações) são mais apropriadas para cada valor de razão sinal-ruído (os estados), o que implica uma convergência mais rápida do algoritmo.

De acordo com os resultados ilustrados na Fig. 3.5(b), não há uma influência significativa

dos valores de ϵ_f no que diz respeito à convergência do algoritmo de aprendizado por reforço, sendo observada pouca diferença entre os três casos apresentados. Ainda assim, é necessário cautela no ajuste desse parâmetro: um valor muito alto de ϵ_f é desejável apenas em cenários de grande variabilidade, em que a capacidade de rastreio (*tracking*) do algoritmo é fundamental para acompanhar as mudanças do ambiente. Este comportamento implica necessariamente em uma menor taxa de explotação da política ótima para uma dada configuração do ambiente.

A Fig. 3.6 exibe a eficiência espectral e a taxa de erro de pacote das abordagens consideradas quando utilizadas em um cenário no qual existe a presença de interferência colorida. Para as simulações realizadas, na entrada do sistema de recepção estava presente, além do ruído térmico, modelado como ruído aditivo branco gaussiano, ou ruído AWGN (*additive white gaussian noise*), um sinal interferente do tipo OFDM de estrutura similar ao encontrado no padrão 3GPP-LTE e cuja potência média era três vezes superior à variância do ruído branco gaussiano.

Exceto para valores muito baixos ou muito altos de razão sinal-ruído-interferência (SINR, *signal to interference plus noise ratio*), existe uma diferença na eficiência espectral de ambas as abordagens que pode chegar a 1 bps/Hz, dependendo da região de SINR considerada, o que representa um ganho de até 40% em termos de eficiência espectral. Nesta situação, uma das limitações da solução do problema de modulação e codificação adaptativas por tabela de consulta (ou outras abordagens de aprendizado supervisionado) é evidenciada: em geral, é muito difícil obter um conjunto de dados que possa representar adequadamente o cenário de estudo, especialmente em cenários de rádio cognitivo, nos quais o comportamento do sinal interferente pode variar imensamente e não é possível seu conhecimento *a priori*. Dessa forma, o desempenho da abordagem por tabela de consulta é comprometido.

Por outro lado, a solução que utiliza aprendizado por reforço é capaz de aprender, de forma *on-line*, as características especificas de um determinado ambiente ou cenário, ajustando de forma apropriada os limiares para a comutação entre os diferentes esquemas de modulação e codificação e, de forma indireta, mantendo a taxa de erro de pacote inferior a 10%. Este fato é posteriormente confirmado pelo resultado exibido na Fig. 3.7, que ainda considera a presença de interferência colorida, porém agora com um sinal interferente cuja potência é oito vezes superior à variância do ruído térmico. Observa-se não apenas uma diferença entre a eficiência espectral alcançada, como também um aumento na taxa de erro de pacote na solução por tabela de consulta.

Investigou-se também a possibilidade de aplicação do algoritmo de aprendizado por reforço de estados contínuos em situações nas quais o canal de comunicação varia com o tempo [86]. A Fig. 3.8(a) mostra a capacidade de rastreio do algoritmo proposto. Para a realização da simulação, o valor da razão sinal-ruído foi mantido em 33 dB. Durante a transmissão dos primeiros 300 quadros, é considerado o cenário em que apenas o ruído brando é adicionado ao sinal que é transmitido. Entre os quadros 301 e 500, além da presença do ruído gaussiano, é adicionado um sinal de interferência colorido, cuja potência é três vezes superior à potência do ruído branco, conforme descrito nos parágrafos anteriores. Por último, para os quadros de 501 a 700, a potência

Tabela 3.3: Características das Imperfeições de RF

Imperfeição	**Valor**
Energia do ruído de fase	0.013 rad^2
Desbalanceamento de fase	3°
Desbalanceamento de amplitude	1.05

da interferência é aumentada para um valor cinco vezes superior à potência do ruído branco.

Na Fig. 3.8, é destacado o tempo de convergência após mudança das características do canal. Na primeira mudança, a convergência ocorre após 50 quadros e, na segunda mudança, a convergência ocorre depois de 30 quadros. Naturalmente, a velocidade de convergência do algoritmo após as mudanças é consideravelmente maior do que sua taxa de convergência inicial, uma vez que não é necessário recalcular a política ótima a partir do zero. A nova política ótima pode ser obtida a partir da atualização dos valores já existentes e calculados da função Q. Apesar da Fig. 3.8(b) mostrar um pequeno aumento da taxa de erro de pacote, este se deve ao comportamento do esquema de modulação e codificação para o cenário em questão. É importante ainda destacar que o valor de $\epsilon_f = 0.05$ foi capaz de prover o grau de exploração necessário do ambiente para que uma nova política fosse obtida neste cenário variante no tempo.

Finalmente, é considerado um cenário em que imperfeições de RF não compensadas são introduzidas no sistema receptor. Mais especificamente, são tratados os casos em que há ruído de fase e desbalanceamento de I/Q no sistema receptor [87]. Os valores utilizados para a simulação são mostrado na Tabela 3.3. Os resultados são mostrados na Fig. 3.9.

Como pode ser observado, a eficiência espectral do sistema é diminuída de forma geral, e o desempenho da abordagem por tabela de consulta é pior do que o desempenho do algoritmo de aprendizado por reforço. Neste, o agente é capaz de aprender que na região de operação em que a razão sinal-ruído é mais elevada, em que a opção mais imediata seria utilizar o esquema de modulação de ordem mais elevada, como o 64QAM, é melhor utilizar uma modulação de ordem mais baixa, pois a presença de imperfeições de RF faz com que a utilização de modulações de ordem mais elevada aumente a taxa de erro de pacote, diminuindo a vazão do sistema como um todo. A tabela de consulta, ao contrário, possui limiares de transição fixos, determinados *a priori* e *off-line*, o que não permite sua posterior adaptação para características particulares de diferentes *front-ends* de RF. Na verdade, em situações práticas, nas quais a variedade de imperfeições pode ser muito grande, torna-se impraticável a obtenção de tabela de consulta especificas para cada caso. Utilizando a abordagem por aprendizado por reforço, a adaptação de enlace pode ser realizada para cada tipo específico de terminal, se assim desejado.

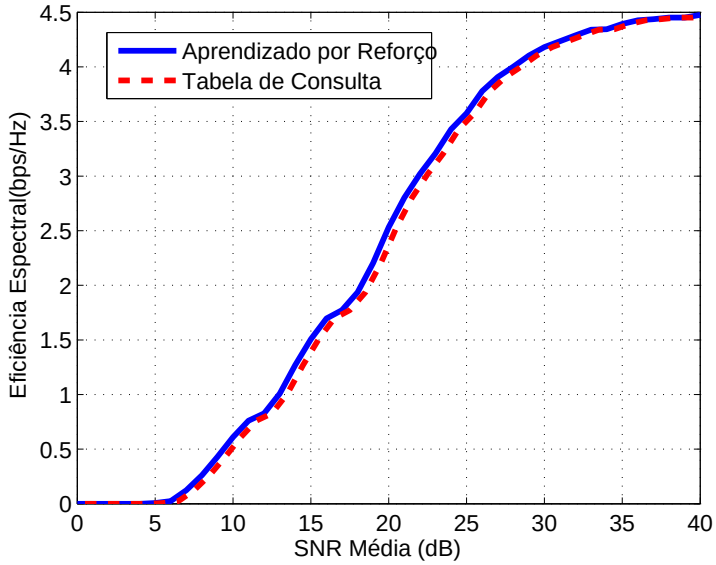
3.5 CONCLUSÃO

Este capítulo apresentou como contribuição a abordagem da técnica de modulação e codificação adaptativas em sistemas OFDM por meio aprendizado por reforço de estados contínuos, cujo objetivo é maximar a eficiência espectral do sistema de comunicação. Utilizou-se uma métrica unidimensional para a medida de qualidade do enlace para identificar o estado do ambiente (o canal de comunicação), e as ações tomadas pelo agente consistiam das combinações permitidas entre os diferentes esquemas de modulação e codificação disponíveis no transmissor. A proposta, diferentemente de outras abordagens que também possuem como base o aprendizado de máquina, não depende de uma etapa de treinamento *off-line*, aprendendo diretamente a partir da interação com o ambiente. O capítulo ainda trouxe como contribuição a proposta de modificação do algoritmo LSPI, introduzindo em sua operação etapas de exploração e de busca da política ótima, de forma a tornar possível sua operação em tempo real.

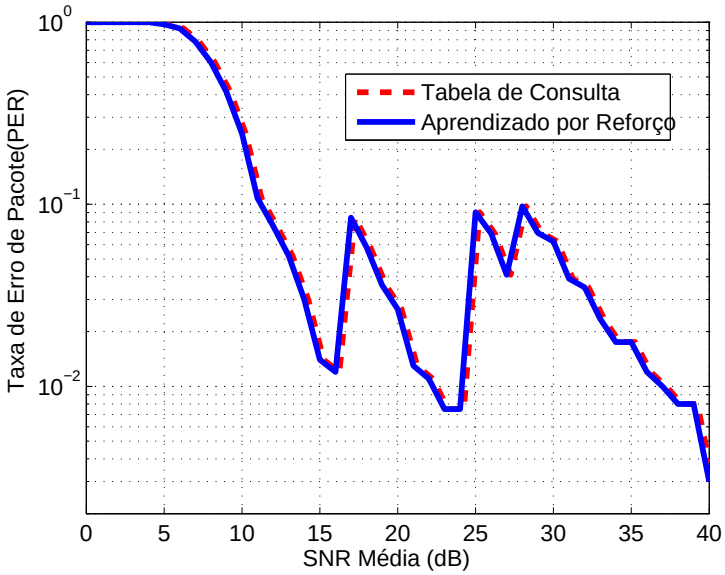
O desempenho da estratégia adotada foi ainda comparado com a solução provida pela tabela de consulta em termos de eficiência espectral alcançada, taxa de erro de pacote e capacidade de rastreio (esta inexistente na solução por tabela de consulta). Constatou-se a superioridade da utilização do aprendizado por reforço em ambientes variantes no tempo ou nos quais o transmissor deve se adaptar a características específicas de diferentes *front-ends* de RF.

O ganho da estratégia de aprendizado por reforço foi condicionado por sua capacidade de adaptação e aprendizado, o que não ocorre com as tabelas de consulta. Em particular, as perdas observadas na abordagem por tabelas de consulta se devem à seleção inapropriada de parâmetros de transmissão, pois estas foram criadas supondo a ausência de imperfeições de RF nos transceptores, e supondo ainda que o ruído que estava presente na entrada do sistema de recepção era do tipo branco e gaussiano. O aprendizado por reforço, ao contrário, não depende da existência de nenhum conjunto de treinamento ou simulações realizadas *a priori*, e não faz nenhuma suposição a respeito do comportamento do canal, sendo capaz de se adaptar e atualizar os parâmetros de transmissão em tempo real e sob condições diferentes de operação do par transmissor-receptor e do enlace de comunicação.

No próximo capítulo, será considerado uma outra forma de adaptação de enlace, que é a alocação de potência em sistemas OFDM. Nesta nova abordagem, também será fornecida uma solução via aprendizado por reforço.



(a) Eficiência espectral



(b) Taxa de erro de pacote

Figura 3.3: Eficiência espectral média e taxa de erro de pacote para o problema de modulação e codificação adaptativas utilizando as abordagens de tabela de consulta e aprendizado por reforço em um cenário macrocelular suburbano.

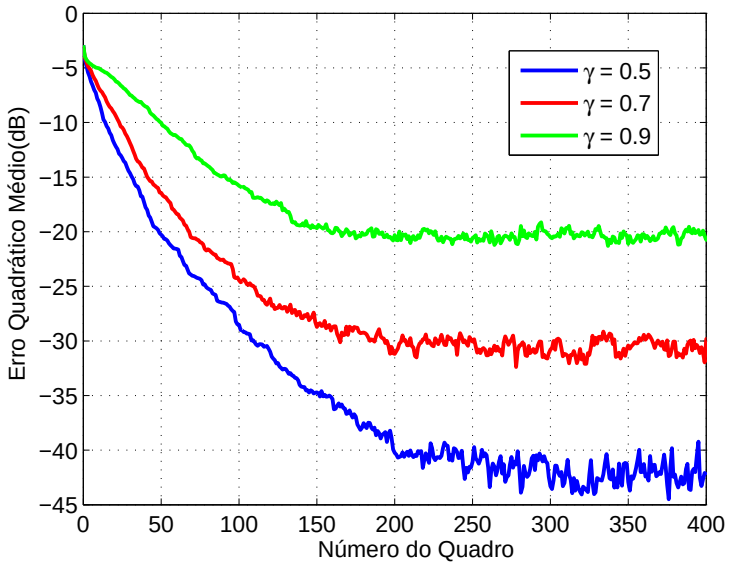
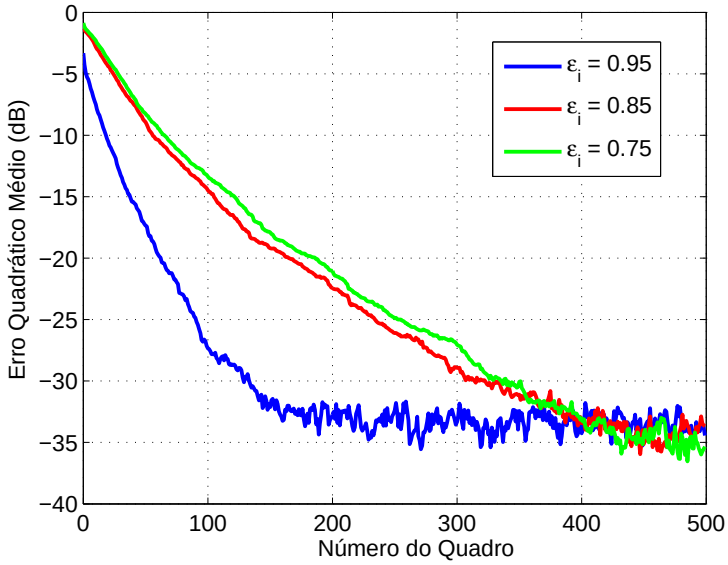
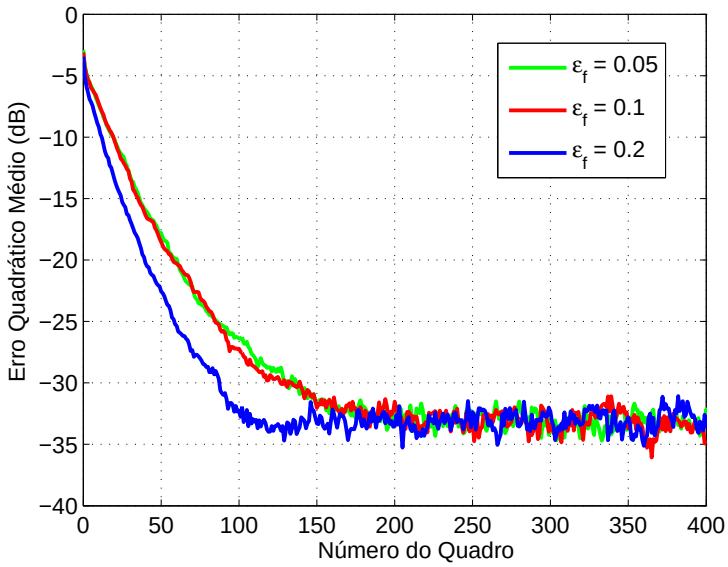


Figura 3.4: Influência do fator de desconto γ na convergência do algoritmo de aprendizado por reforço para $\epsilon_i = 0.95$ e $\epsilon_f = 0.05$.

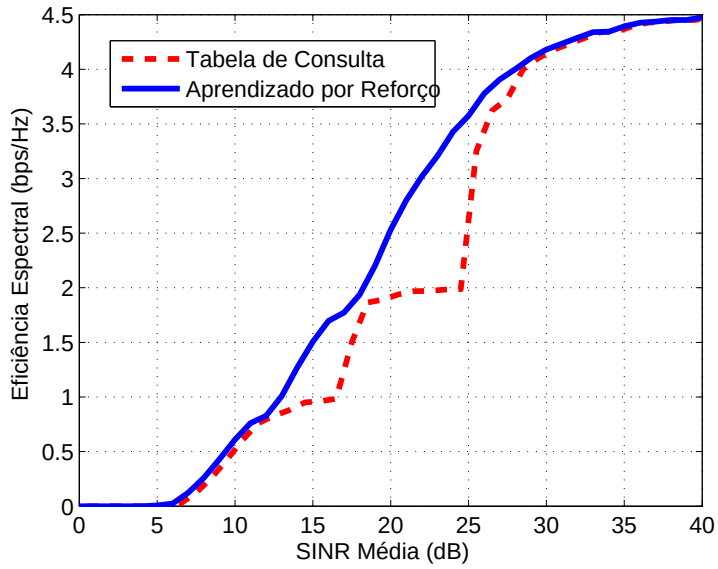


(a) Comportamento da convergência para diferentes valores de ϵ_i

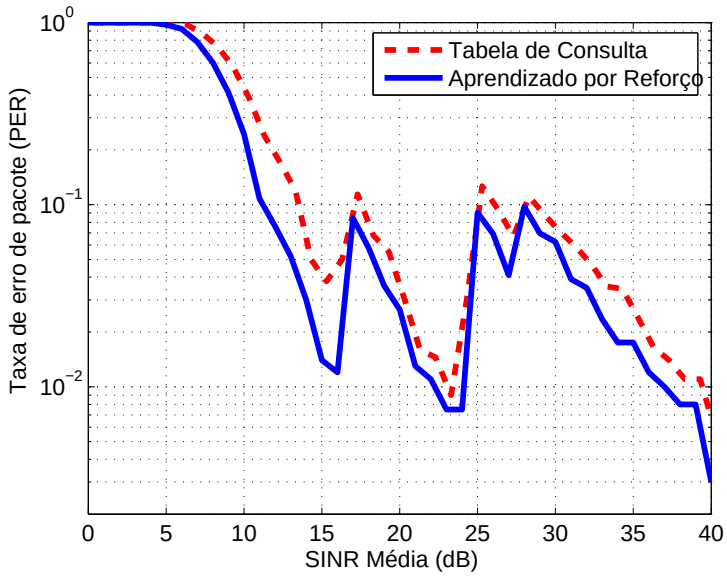


(b) Comportamento da convergência para diferentes valores de ϵ_f

Figura 3.5: Influência da escolha dos parâmetros ϵ_i e ϵ_f da política de exploração ϵ-*greedy* na convergência do algoritmo de aprendizado por reforço. Nas situações mostradas, manteve-se $\gamma = 0.65$.

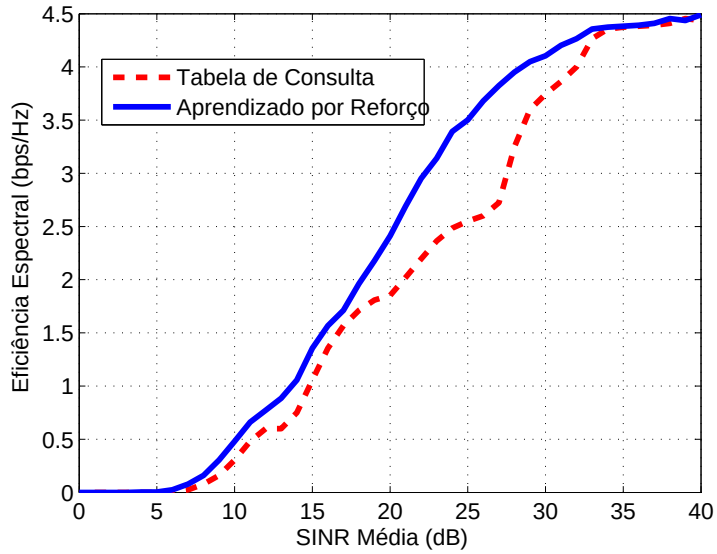


(a) Eficiência espectral

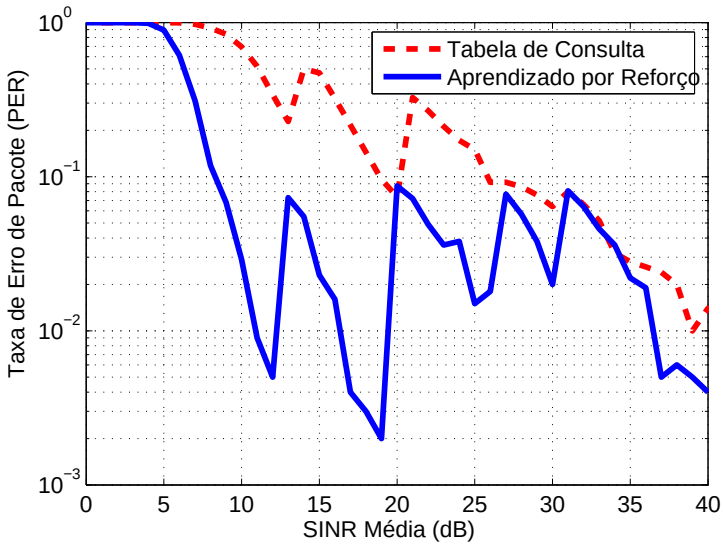


(b) Taxa de erro de pacote

Figura 3.6: Eficiência espectral média e taxa de erro de pacote para a estratégia de modulação e codificação adaptativas utilizando a abordagem por tabela de consulta e aprendizado por reforço em um cenário macrocelular suburbano com interferência colorida. A potência do sinal interferente é três vezes superior à variância do ruído branco gaussiano.

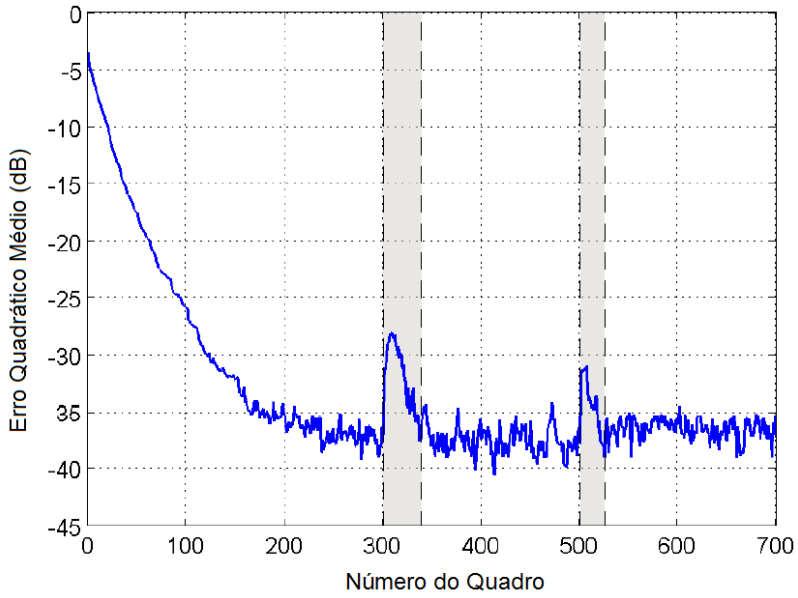


(a) Eficiência espectral

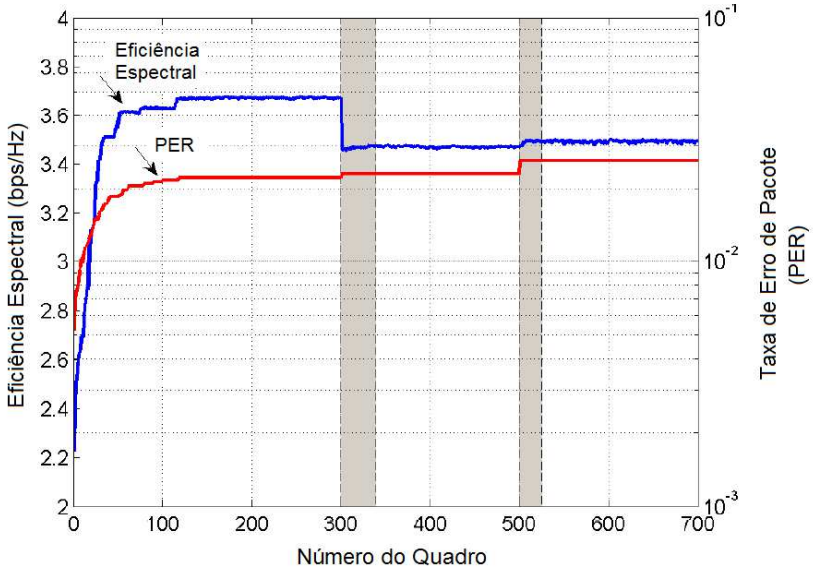


(b) Taxa de erro de pacote

Figura 3.7: Eficiência espectral média e taxa de erro de pacote para a estratégia de modulação e codificação adaptativas utilizando a abordagem por tabela de consulta e aprendizado por reforço em um cenário macrocelular suburbano com interferência colorida. A potência do sinal interferente é oito vezes superior à variância do ruído branco gaussiano.

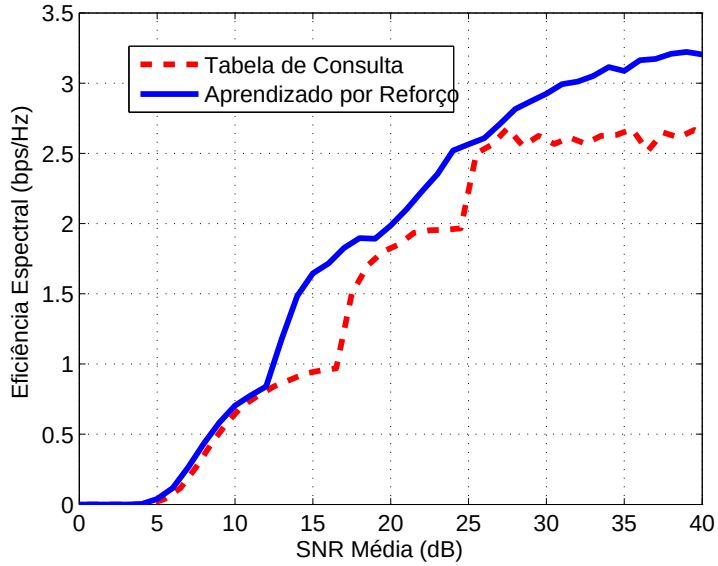


(a) Comportamento da convergência

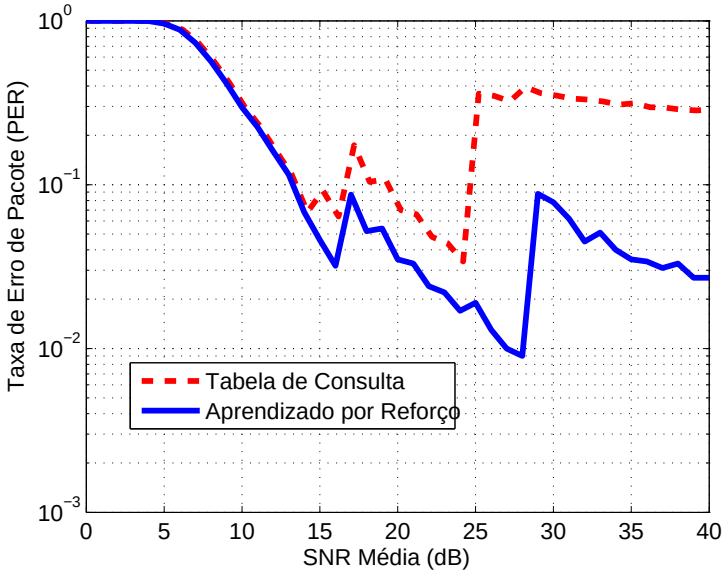


(b) Eficiência espectral e taxa de erro de pacote

Figura 3.8: Capacidade de rastreio do algoritmo de aprendizado por reforço em um cenário cujo canal de comunicação é variante no tempo. A SINR foi mantida fixa em um valor de 33 dB.



(a) Eficiência espectral



(b) Taxa de erro de pacote

Figura 3.9: Eficiência espectral média e taxa de erro de pacote das técnicas de aprendizado por reforço e tabela de consulta em um cenário suburbano macrocelular, considerando a presença de imperfeições de RF não compensadas no receptor.

4 IMPLEMENTAÇÃO DA ESTRATÉGIA DE BIT LOADING UTILIZANDO APRENDIZADO POR REFORÇO

4.1 INTRODUÇÃO

Conforme mostrado no Capítulo 3, uma das possíveis formas de se melhorar o desempenho de sistemas de comunicação é por meio de estratégias de adaptação dos parâmetros do sistema às condições de transmissão e de recepção. Mais especificamente, o capítulo anterior tratou da estratégia de modulação e codificação adaptativas, em que a constelação utilizada para o processo de modulação e a taxa de codificação de canal podiam variar de acordo com as condições do canal de comunicação, isto é, quanto melhor a resposta do canal, maior a ordem de modulação utilizada e maior a taxa de código.

Uma outra forma de prover adaptação em sistemas multiportadora é por meio da estratégia de *bit loading*, que consiste em permitir que cada subportadora utilize um tipo de modulação diferente, de acordo com a resposta de canal que esta experimenta. A busca de um determinado esquema de alocação de *bits* entre as subportadoras depende do objetivo a ser alcançado (minimizar a potência alocada, maximizar a taxa de transmissão etc.) e das restrições impostas à solução (potência máxima disponível, potência média utilizada etc.). São dois os problemas típicos encontrados na literatura [88]: o problema da maximização de taxa de transmissão (*rate maximization*) e o problema de maximização de margem (*margin maximization*) ou minimização de potência. Nas próximas seções, será considerado o problema de maximização de taxa, dada a busca que existe por taxas de transmissão mais elevadas em sistemas de comunicação sem fio.

Diferentemente do que se expõe na literatura clássica e tradicional de alocação de potência, que se baseia em expressões analíticas e otimização gulosa, este capítulo apresenta a solução para o problema de maximização de taxa em sistemas OFDM utilizando a estratégia de *bit loading* implementada por meio do algoritmo *Q-Learning*, este já tradicionalmente utilizado na solução de problemas de aprendizado por reforço. A principal contribuição deste capítulo consiste na modelização e na solução do problema de *bit loading* em sistemas multiportadora por meio do *framework* de aprendizado por reforço, utilizando algoritmos e estratégias de exploração já conhecidas na literatura de aprendizado de máquina, mas anteriormente inexplorados para este problema específico.

Inicialmente, é apresentada a solução clássica para o problema de alocação de potência, dada pelo algoritmo *water-filling*. Em seguida, detalha-se o algoritmo de Levin-Campello, que provê

a alocação ótima de *bits* no problema de *bit loading* discreto. Posteriormente, é apresentada a modelização do problema de alocação de *bits*, de forma a obter uma solução via aprendizado por reforço, expondo os estados, ações e recompensas disponíveis ao agente inteligente. Finalmente, são mostrados o cenário de simulação e os resultados obtidos, comparando os desempenhos da solução proposta e do algoritmo de Levin-Campello, tomando como referência a situação para a qual não é utilizada qualquer estratégia de adaptação de enlace.

4.2 OTIMIZAÇÃO WATER-FILLING

Da Teoria de Informação, a capacidade $\overline{c}$ de um canal do tipo AWGN, medida em *bits* por dimensão de transmissão, é dada por [89]

$$\overline{c} = \frac{1}{2}\log_2\left(1 + SNR\right) \tag{4.1}$$

em que SNR é a razão sinal-ruído do canal de comunicação.

Entretanto, a capacidade de canal obtida por meio da Eq. (4.1) é teórica. Em sistemas reais, o número de *bits* de informação que é efetivamente transmitido é menor do que aquele dado pela fórmula de capacidade do canal. Para quantificar o hiato que existe entre a capacidade teórica e a taxa efetivamente alcançada, é introduzida a formulação de *gap* de capacidade [90]. A taxa real de transmissão normalizada pela largura de banda utilizada pelo sistema, $\overline{b}$, é dada por

$$\overline{b} = \frac{1}{2}\log_2\left(1 + \frac{SNR}{\Gamma}\right) \tag{4.2}$$

em que Γ é um valor de *gap* introduzido para quantificar a diferença de desempenho entre o limite teórico e o observado na prática. Fixadas a modulação, a codificação e a probabilidade de erro de *bit*, a Eq. (4.2) pode ser reescrita como

$$\Gamma = \frac{SNR}{2^{2\overline{b}} - 1} \tag{4.3}$$

Quanto maior o valor do *gap* Γ, pior é o desempenho do sistema quando comparado ao limiar teórico fornecido pela Eq. (4.1). O valor de Γ também depende da região de taxa de erro de *bit* na qual o sistema opera, e seu valor aumenta conforme a taxa de erro de *bit* diminui. Por exemplo, para modulação QAM e BER de 10^{-6}, $\Gamma \approx 8,8$ dB. Para uma BER de 10^{-7}, tem-se $\Gamma \approx 9,5$ dB [90]. Naturalmente, a utilização de codificação de canal é capaz de reduzir o valor de Γ.

Em sistemas OFDM, um canal que apresenta seletividade em frequência é particionado em N_c subcanais banda estreita paralelos, representados por cada uma das subportadoras utilizadas. Logo, o número total de *bits* que pode ser transmitido é dado por [31]

$$b = \frac{1}{2}\sum_{i=1}^{N_c} \log_2\left(1 + \frac{E_i|H_i|^2}{\Gamma\sigma^2}\right) \tag{4.4}$$

em que E_i é a energia alocada na i-ésima subportadora e $|H_i|$ é o ganho de canal na i-ésima subportadora. O valor de σ^2 é a variância do ruído branco, cujo valor é o mesmo em todas as N_c subportadoras.

Para o problema de maximização de taxa, busca-se encontrar quais são os valores de E_i que maximizam a taxa de transmissão. Em outras palavras, deve-se maximizar o valor de b na Eq. (4.4) sujeito à restrição

$$\sum_{i=1}^{N_c} E_i = N_c\overline{E} \tag{4.5}$$

com $\overline{E}$ representando a energia média alocada em cada subcanal. Deve-se determinar a energia que deve ser alocada em cada subportadora de tal forma que a maior taxa de transmissão possível seja alcançada, e que a energia utilizada seja limitada superiormente pela energia total disponível ao sistema de transmissão.

Mostra-se que a maximização da Eq. (4.4), com a restrição dada pela Eq. (4.5), equivale à solução de um sistema linear de $N_c + 1$ equações e $N_c + 1$ incógnitas, dado por [31]:

$$\begin{aligned}
E_1 + \frac{\Gamma\sigma^2}{|H_1|} &= K \\
E_2 + \frac{\Gamma\sigma^2}{|H_2|} &= K \\
E_3 + \frac{\Gamma\sigma^2}{|H_3|} &= K \\
\vdots &= \vdots \\
E_1 + E_2 + E_3 + \cdots + E_{N_c} &= N_c\overline{E}
\end{aligned} \tag{4.6}$$

em que K é uma constante.

A solução do sistema deve ser obtida de forma recursiva [31], pois é possível que alguns valores de E_i obtidos pela solução direta do sistema sejam negativos, originando um resultado que não possui sentido físico. Esta solução recursiva é dada como segue: em primeiro lugar, os subcanais são ordenados de forma decrescente de acordo com o valor do ganho $|H_i|^2$, e o sistema resultante é resolvido para todos os valores de E_i. Caso exista algun $E_i < 0$, deve ser eliminada a equação n que possui o menor valor de $|H_n|$, e fazer $E_n = 0$. O conjunto restante de equações deve ser resolvido seguindo o mesmo procedimento, até que $E_i \geq 0\ \forall i \in \{1, 2, \cdots, N_c\}$.

Denominando por N_c^* o número de equações que produz a solução do sistema dado pela Eq. (4.6) sem qualquer valor negativo de energia, a constante K é dada por [31]:

$$K = \frac{1}{N_c^*}\left[\overline{E} + \Gamma\sigma^2 \sum_{i=1}^{N_c^*} \frac{1}{|H_i|^2}\right] \tag{4.7}$$

de tal forma que a alocação final em cada subcanal é dada por

$$\begin{aligned} E_i &= K - \frac{\Gamma\sigma^2}{|H_i|^2} \\ b_i &= \frac{1}{2}\log_2\left(\frac{K|H_i|^2}{\Gamma\sigma^2}\right) \end{aligned} \tag{4.8}$$

considerando $i = 1, 2, \cdots, N_c^*$.

A solução dada pela Eq. (4.8) é clássica para o problema de maximização de taxa, e é denominada *water-filling*, e mostra-se que ela é única por se tratar de um problema de otimização convexo [88].

O termo *water-filling* (enchimento com água) surge a partir da representação gráfica da solução, conforme mostra a Fig. 4.1. O conjunto de valores $1/|H_i|^2$ pode ser visto como o fundo de um reservatório, que deve ser enchido com $N_c\overline{E}$ unidades de água, de tal forma que o nível de água não ultrapasse o valor K. Quanto melhor a resposta do canal (ou seja, maior $|H_i|^2$), mais potência é alocada pois mais fundo é o reservatório. Quanto maior a perda do canal (menor $|H_i|^2$), menor a potência alocada, podendo esta ser nula, dependendo da quantidade de água (energia) disponível.

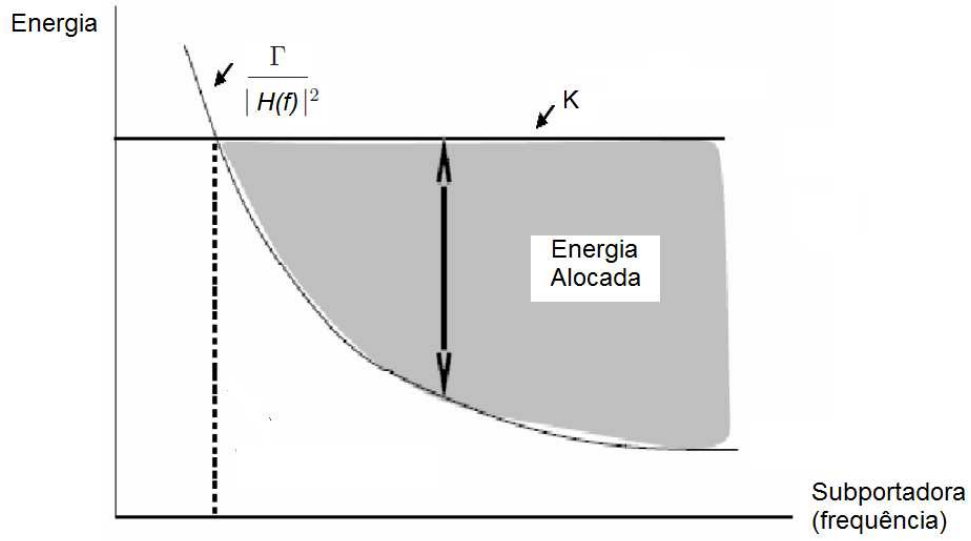


Figura 4.1: Visão intuitiva da lógica utilizada pelo algoritmo *water-filling* para a solução do problema de maximização de taxa.

Uma segunda ilustração mostrando a alocação para o caso em que potência é alocada em 6 subportadoras é mostrada na Fig. 4.2. Devido às restrições impostas para esta situação, nenhuma potência foi alocada à terceira subportadora.

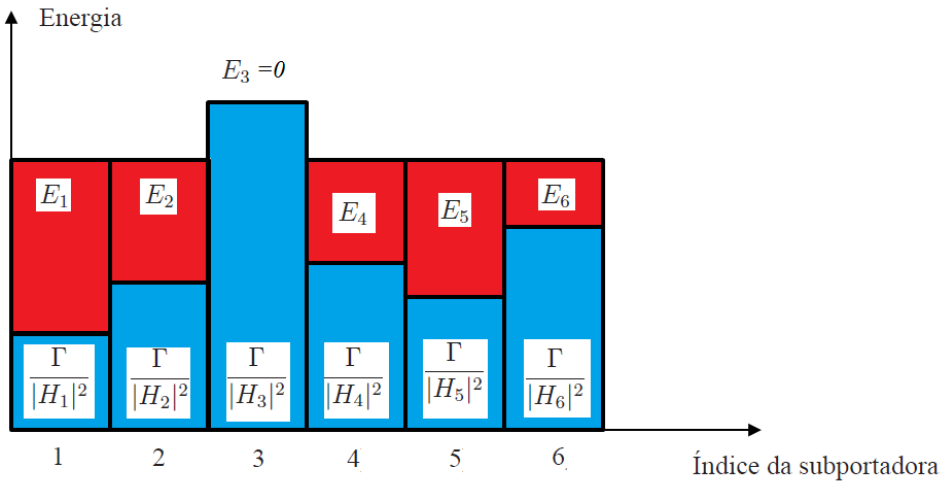


Figura 4.2: Ilustração do resultado do algoritmo de *water-filling* para um cenário com seis subportadoras.

4.3 BIT LOADING DISCRETO

A solução obtida por meio do algoritmo de *water filling* pressupõe que os *bits* alocados em uma determinada subportadora podem assumir qualquer valor real não negativo, conforme mostra a Eq. (4.8). Entretanto, no projeto de sistemas reais, esta não é uma hipótese realista. A quantidade de *bits* disponíveis a serem alocados pertence a um conjunto finito de valores, que dependem das diferentes técnicas de modulação e de codificação que estão disponíveis nos sistemas de transmissão e de recepção. Em outras palavras, o número de *bits* que pode ser alocado em uma subportadora não é contínuo, mas sim discreto. Logo, a alocação de *bits* está sujeita à granularidade (ou passo de quantização) β dos esquemas de modulação e codificação (o valor β depende dos esquemas de modulação e codificação disponíveis, e representa a diferença observada na eficiência espectral entre os sucessivos esquemas de modulação e codificação). Este problema é conhecido como *bit loading* discreto, e sua solução será considerada nesta seção.

É importante destacar que, nos últimos anos, foram propostas várias técnicas diferentes para a solução do problema de *bit loading* discreto em sistemas multiportadora. As primeiras soluções [91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101] buscavam por estratégias gulosas (*greedy*), que são computacionalmente complexas, ou por soluções subótimas, baseadas na aproximação do problema de *bit loading* discreto por problemas de otimização convexa. Mais recentemente, são propostas soluções extremamente específicas para os cenários tratados, como, por exemplo, a presença de múltiplas antenas [102], o caso monousuário ou multiusuário [103] e até mesmo o tipo de codificadores de canal que é utilizado no sistema OFDM [104].

Este trabalho fará a exposição do algoritmo de Levin-Campello [105, 106] por duas razões: Campello foi o primeiro autor que desenvolveu um *framework* matematicamente rigoroso, embasado na teoria de otimização discreta, para tratar o problema de *bit loading* discreto,

permitindo inclusive verificar a otimalidade da solução encontrada. Em segundo lugar, a maioria das propostas mais recentes de algoritmos para a alocação de *bits* utiliza como etapa intermediária o algoritmo proposto por Campello, tornando sua abordagem original ainda muito utilizada.

4.3.1 Algoritmo de Levin-Campello

A essência do algoritmo resume-se ao fato de que uma busca sequencial gulosa (*greedy*) conduz à uma alocação discreta de *bits* que é globalmente ótima [107, 108, 109]. Logo, para encontrar a solução ótima, os *bits* são alocados de forma sequencial, iniciando sempre na subportadora que exige a menor quantidade de energia incremental para a transmissão, até que a restrição quanto à potência máxima alocada tenha sido atingida [110].

Nos próximos parágrafos, serão abordados os conceitos utilizados por Campello e a operação de seu algoritmo.

Será denotado por $E_i(b_i)$ a energia utilizada ao alocar b_i *bits* na subportadora i. A energia incremental de uma subportadora i para a alocação b_i, representada por $\Delta E_i(b_i)$, é definida como:

$$\Delta E_i(b_i) = E_i(b_i) - E_i(b_i - \beta) \tag{4.9}$$

e representa a energia que deve ser adicionada à subportadora i para que, em vez de carregar $b_i - \beta$ *bits*, passe a carregar b_i *bits*.

O algoritmo de Levin-Campelo é dividido em duas etapas, denominadas EF (*Efficientizing*) e ET (*E-tightening*). A primeira etapa, EF, consiste em, a partir de uma alocação inicial de *bits* qualquer, obter uma alocação eficiente.

Uma alocação é dita eficiente, para um dado β, se satisfaz a condição

$$\max_i \Delta E_i(b_i) \leq \min_j \Delta E_j(b_j + \beta) \tag{4.10}$$

ou seja, não é possível reduzir a energia utilizada por meio da realocação de *bits* entre as subportadoras, mantendo-se constante o número total de *bits* alocados.

Um fluxograma da primeira etapa do algoritmo é mostrado na Fig. 4.3. Em cada iteração da etapa EF, busca-se pela subportadora que demanda a menor quantidade de energia para aumentar a alocação em β *bits*. Em seguida, busca-se pela subportadora que devolve a maior quantidade de energia para o sistema quando desta se removem β *bits*. Por fim, os β *bits* são realocados na subportadora que exige a menor quantidade de energia para a alocação de β *bits*. Dessa forma, a distribução da função de energia alocada será uma função monotonicamente crescente com o número de *bits* alocados (como ocorre com sistemas práticos).

A segunda parte do algoritmo, ET, consiste em, a partir de uma alocação eficiente de *bits*,

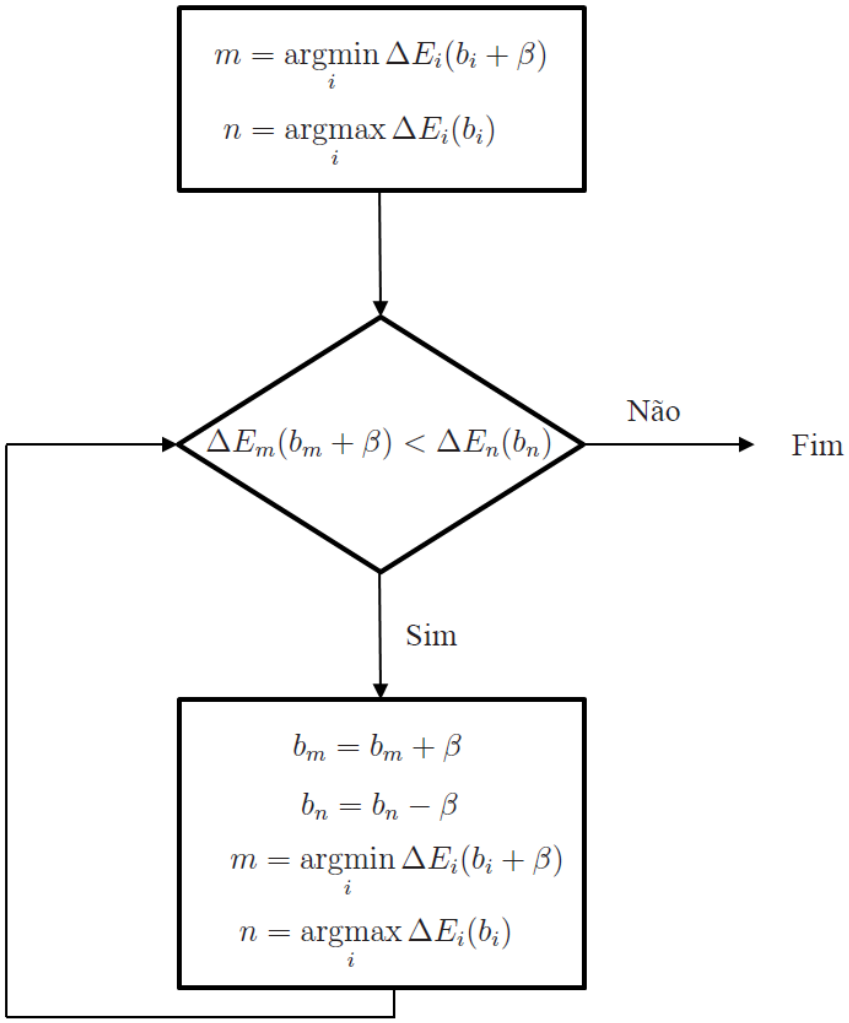


Figura 4.3: Fluxograma da etapa EF do algoritmo de Levin-Campello para a obtenção de uma alocação eficiente de *bits*.

obter uma alocação *E-tight*, ou seja, reduzir o número de *bits* alocados quando a energia total alocada ultrapassa o limite permitido. Uma alocação de *bits* é dita *E-tight* se satisfaz à condição

$$0 \leq N\overline{E} \leq \sum_{i=1}^{N_c} \Delta E_i(b_i) + \min_{1 \leq i \leq N_c} \Delta E_i(b_i + \beta) \tag{4.11}$$

Ou seja, se uma alocação satisfaz a Eq. (4.11), é impossível adicionar β *bits* em qualquer subportadora sem que a restrição quanto à potência alocada continue sendo satisfeita.

Um fluxograma desta etapa é mostrado na Fig. 4.4. A cada iteração, se a energia total alocada é maior do que a restrição imposta, o algoritmo identifica a subportadora que requer a maior quantidade de energia para adicionar os últimos β *bits*, e remove β *bits* dessa subportadora. Se a nova energia de símbolo é menor do que a restrição de potência e é possível ainda alocar β

bits, então esses *bits* são alocados para a subportadora que exige a menor energia incremental. O processo continua até que se obtenha uma alocação *E-tight*.

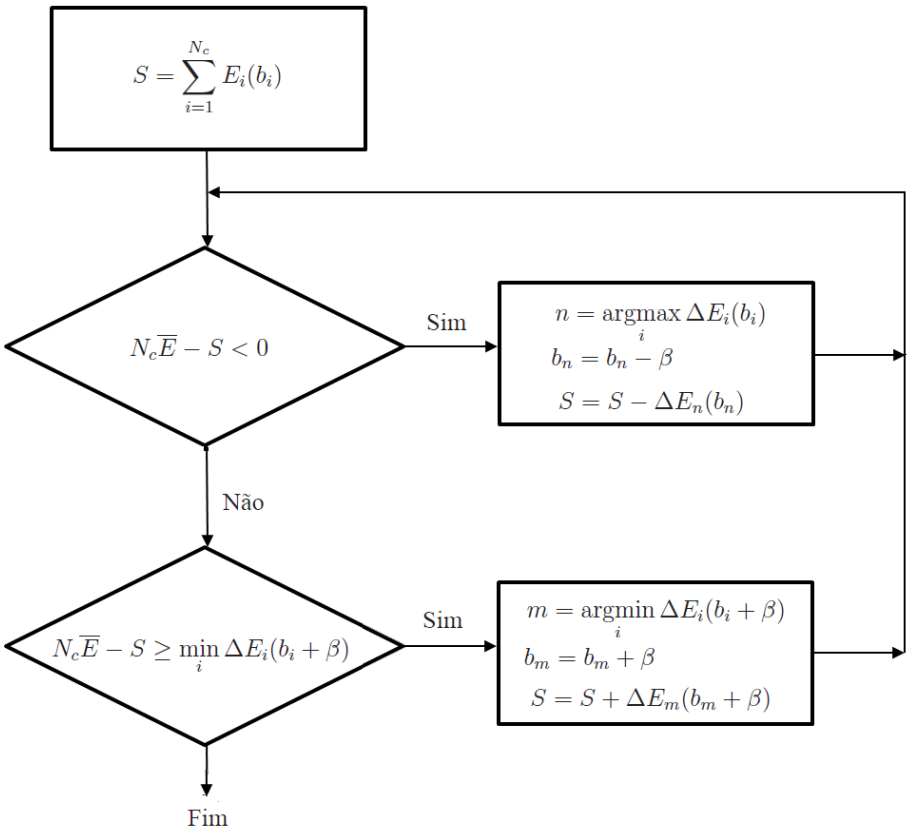


Figura 4.4: Fluxograma da etapa ET do algoritmo de Levin-Campello para a obtenção de uma alocação *E-tight*.

Em [105], mostra-se que, ao aplicar a etapa ET a uma alocação eficiente, tem-se a solução para o problema do *bit* loading discreto para a maximização de taxa.

4.4 SOLUÇÃO DO PROBLEMA DE BIT LOADING DISCRETO POR MEIO DE APRENDIZADO POR REFORÇO

Conforme mostrado nas Seções 4.2 e 4.3, a solução para o problema de alocação de *bits* depende do conhecimento do valor de *gap* Γ. Este é função do tipo de modulação, codificação, taxa de erro desejada na operação do sistema, do ganho de canal e do tipo de ruído presente no sistema receptor. Na maior parte dos casos, este valor é de difícil obtenção analítica, quer seja pela complexidade imposta pelos sistemas OFDM, quer seja pelas hipóteses de que os transceptores utilizados são ideais e que o ruído introduzido no sistema de comunicação é branco e gaussiano.

Uma outra forma de obtenção do valor de *gap* é por meio de simulações computacionais, realizadas antes do início de operação do sistema, ainda na fase de projeto. Entretanto, o número de simulações a serem realizadas é potencialmente muito grande, pois deve refletir as diferentes condições de operação do sistema de comunicação, como as imperfeições de RF presentes nos sistemas de transmissão e de recepção, ou à própria variação do comportamento do ruído e da interferência presente no canal de comunicação.

Uma das formas de se contornarem as limitações consideradas e apresentadas é por meio técnicas de aprendizado de máquina. Como ponto inovador, é apresentado nesta seção o tratamento do problema de *bit loading* discreto utilizando o *framework* de aprendizado por reforço. São também expostos os resultados de simulação comparando a solução proposta com a obtida pela utilização do algoritmo de Levin-Campello. Na solução proposta, não é necessário o conhecimento do valor de Γ, uma vez que este será conhecido de forma implícita por meio da solução do problema de AMC, conforme considerado no Capítulo 3.

4.4.1 Modelo do Sistema

O modelo do sistema é similar ao utilizado no Capítulo 3. Tem-se um sistema de comunicação que utiliza OFDM nos padrões similares ao LTE, com banda de transmissão disponível de 1,4 MHz, o que resulta em 72 subportadoras disponíveis, agrupadas em 6 *resource blocks* [111], identificados por RB_1, RB_2, RB_3, RB_4, RB_5 e RB_6. Uma vez que, no padrão, a alocação de recursos não é realizada por subportadora, mas sim por *resource block*, que representarão, para esta modelagem, os subcanais banda estreita nos quais o *bit loading* deve ser realizado. O transmissor deve então realizar a alocação de *bits* em um conjunto de até 6 *resource blocks* disponíveis [111].

A transmissão dos *bits* de informação é feita por meio de pacotes de dados. A cada pacote, é adicionado um campo de CRC e, em seguida, toda essa informação é apresentada à entrada de um codificador de canal do tipo convolucional. Os *bits* codificados são modulados e o sinal resultante é utilizado para a formatação dos símbolos OFDM, que serão acomodados em *resource blocks*. É ainda inserido um intervalo de guarda apropriado em cada símbolo OFDM de forma a compensar a interferência inter-simbólica introduzida pelos múltiplos percursos do canal de comunicação sem fio.

De acordo com as condições de canal de cada *resource block*, o agente deve realizar a alocação de *bits* de forma a maximizar a eficiência espectral. Um diagrama de blocos que ilustra a operação do sistema é mostrada na Fig. 4.5.

Os esquemas de modulação e codificação disponíveis para a realização do *bit loading* discreto são mostrados na Tabela 4.1, assim como suas respectivas eficiências espectrais máximas.

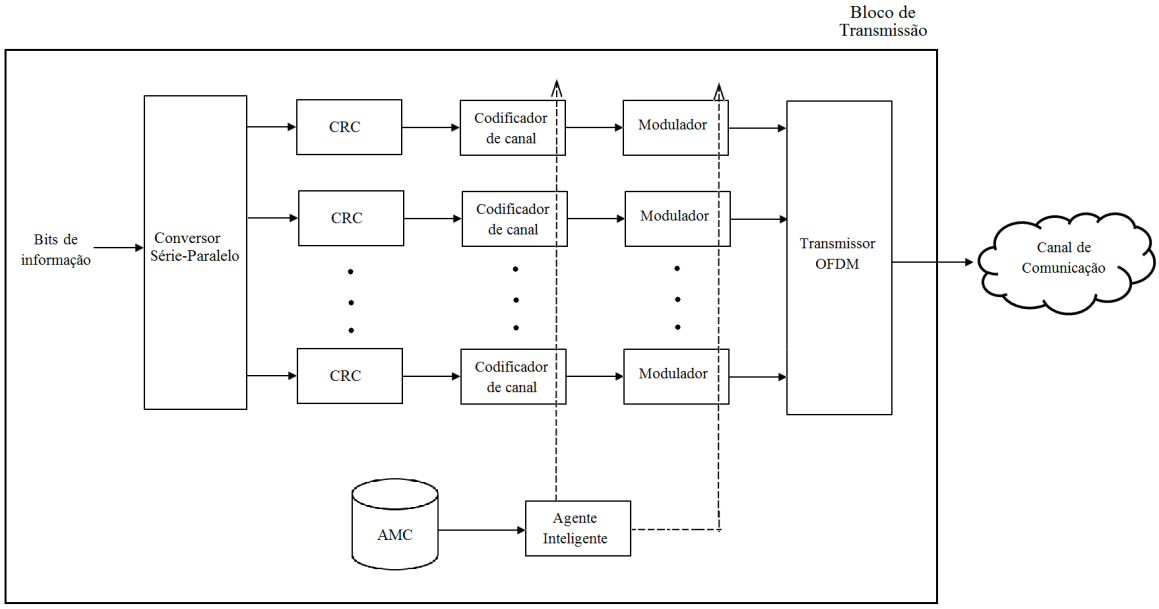


Figura 4.5: Diagrama de blocos do sistema de transmissão que utiliza a solução por aprendizado por reforço para o problema de *bit loading*.

Tabela 4.1: Esquemas de Modulação e Codificação Disponíveis

Número do esquema	**Modulação**	**Taxa de codificação**	**Eficiência Espectral (bps/Hz)**
MCS_1	4QAM (QPSK)	1/2	1
MCS_2	4QAM (QPSK)	3/4	1,5
MCS_3	16QAM	1/2	2
MCS_4	16QAM	3/4	3
MCS_5	64QAM	2/3	4
MCS_6	64QAM	3/4	4,5

4.4.2 Estados, Ações e Recompensas

Os estados são dados pela 7-upla:

$$s = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & P_r \end{bmatrix} \tag{4.12}$$

em que b_i representa a quantidade de *bits* já alocada no i-ésimo *resource block*, com os valores de $i = 1, 2, \cdots, 6$, e os descritores b_i são organizados em ordem crescente de ganho de canal para os *resource blocks*, ou seja, descritor de estados é organizado de tal forma que b_1 se refere à alocação de *bits* no *resource block* que possui o menor ganho de canal, e b_6 se refere à alocação de *bits* no *resource block* que possui o maior ganho de canal. Este processo de ordenação é necessário para evitar qualquer ambiguidade na representação dos estados do ambiente. Conforme descrito na seção 4.3, para a alocação de *bits* entre as subportadoras, o mais importante a ser determinado

é o ganho relativo entre as subportadoras ou *resoure blocks*, e não a posição absoluta que estes ocupam. Logo, a representação de estados procura preservar esta ordem relativa entre os *resoure blocks*.

O valor P_r representa, em porcentagem, a energia disponível ao sistema ao utilizar a alocação dada pela combinação $\begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \end{bmatrix}$. Optou-se por trabalhar com uma abordagem completamente discreta para o problema. Neste caso, os valores de P_r foram quantizados com um passo de 5%, ou seja, $P_r \in \{0\%, 5\%, 10\%, \ldots, 100\%\}$. Cumpre salientar que, pela própria natureza do problema de *bit loading* discreto, os valores de b_i pertencem a um alfabeto finito, sendo, portanto, discretos. Cabe lembrar que o armazenamento dos valores P_r é importante para o problema de alocação pois, como não se está armazenando a informação do ganho de canal, é necessário algum descritor para representar a quantidade de potência já alocada, visto que as outras dimensões da representação de estados armazena apenas a quantidade de *bits* já alocados. Este descritor reflete a diferença de ganho que há entre os *resource blocks* ordenados considerando-se diferentes realizações de canal.

A ação é definida pela dupla:

$$a = \begin{bmatrix} MCS_m & RB_i \end{bmatrix} \tag{4.13}$$

que representa a utilização do MCS_m no i-ésimo *resource block*. Ainda é possível a ação que especifica que nenhuma alteração deve ser feita à alocação de *bits* atual do sistema.

A recompensa do agente é medida pela razão:

$$R(s,a) = \begin{cases} \frac{1}{P_{alocada}} \sum_{i=1}^{6} b_i, & \text{se } P_{alocada} \leq P_{disponivel} \\ -10, & \text{caso contrário} \end{cases} \tag{4.14}$$

em que $P_{alocada}$ corresponde à potência consumida pela alocação $\begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \end{bmatrix}$.

A proposta de função de recompensa busca maximizar a eficiência espectral da alocação de *bits*, ao mesmo tempo que procura manter a potência consumida abaixo do limite disponível. Para alocações que possuem a mesma eficiência espectral, é preferida aquela que utiliza a menor quantidade de energia, o que justifica a presença da potência alocada no denominador da função.

Caso a alocação de *bits* ultrapasse a quantidade de energia disponível, o agente é penalizado com o valor de -10 em sua recompensa. Este valor foi escolhido como uma solução de compromisso, de forma a indicar para o agente a penalização que será recebida caso não realize uma alocação que respeite o limite de potência imposto pelo sistema. A utilização de valores negativos muito próximos de zero faz com que o agente opte excessivamente por ações que ultrapassam a potência máxima disponível para a alocação. Em contrapartida, valores muito menores do que -10 não permitem a exploração apropriada do ambiente, fazendo com que o agente seja excessivamente penalizado.

Cabe ainda observar que o valor de $P_{alocada}$ é determinado de acordo com os limiares de modulação e codificação adaptativas aprendidos no problema de AMC, de forma que a solução do problema de *bit loading* discreto depende primeiramente da solução do problema de AMC, determinado no Capítulo 3, e conforme é explicitado na Fig. 4.5.

Diferentemente do apresentado nos Capítulos 2 e 3, optou-se pela utiização da estratégia de exploração de Boltzmann [47], em que a probabilidade de selecionar uma ação a dado que o ambiente se encontra no estado s, denotada por $P(a|s)$, é dada por

$$P(a|s) = \frac{e^{Q(s,a)/\tau}}{\sum_{a' \in \mathcal{A}} e^{Q(s,a')/\tau}} \tag{4.15}$$

em que τ é denominada temperatura de exploração [44]. Valores maiores de τ fazem com que cada ação seja selecionada com a mesma probabilidade. Valores menores de τ faz com que a política de exploração se aproxime do comportamento guloso [47].

A escolha da estratégia de Boltzmann é justificada ao observar a forma da Eq. (4.15). Conforme os valores da função valor-ação $Q(s, a)$ são atualizados de forma iterativa, a tendência é que as ações com maior valor de Q para um dado estado sejam escolhidas com maior frequência do que as que apresentam menor valor de Q, favorecendo a explotação do ambiente e aproximando-se da política gulosa. Entretanto, cabe observar que outras estratégias de exploração poderiam ser utilizadas, sem comprometimento da convergência do algoritmo.

O algoritmo de aprendizado por reforço utilizado para a solução do problema é o *Q-learning*, que se encontra resumido no Algoritmo 4.

Algoritmo 4 *Q-Learning* para alocação de potência

1. **Repita** até que seja encontrado um estado terminal
   2. O agente percebe o estado atual do ambiente, s
   3. Uma ação a é realizada com probabilidade $P(a|s)$ dada pela Eq. (4.15)
   4. Como resultado da ação, é gerado um sinal de recompensa $R(s, a)$ e o sistema é levado para o estado s'
   5. Q $(s, a) \leftarrow Q(s, a) + \alpha \left[R(s, a) + \gamma \max_{a'} Q(s', a') - Q(s, a)\right]$
7. **Fim**

4.4.3 Parâmetros de Simulação

Transmissor quanto receptor utilizam apenas uma antena. As combinações permitidas entre as diferentes formas de modulação de codificação são mostradas na Tabela 4.1. O código corretor de erros é do tipo convolucional de taxas 1/2, 2/3 ou 3/4. O codificador de canal base é de taxa 1/2 e polinômios geradores dados por $[133, 171]$ (em octal), e as taxas de 2/3 ou 3/4 são obtidas

por meio da perfuração desse código base.

Utilizou-se um modelo de canal multipercursos variante no tempo denominado SCM (*Spatial Channel Model*), que gera os coeficientes de canal de acordo com o modelo padronizado pelo 3-GPP [82, 83]. Os parâmetros ajustados desse modelo de canal encontram-se na Tabela 4.2.

Tabela 4.2: Parâmetros do Canal SCM.

Parâmetro	**Valor**
Frequência de transmissão	2.0 GHz
Velocidade do terminal móvel	10.8 m/s
Número de antenas na estação base	1
Número de antenas na estação móvel	1
Cenário	macrocélula suburbana
Número de multipercursos	19

Quanto ao ajuste do algortimo *Q-Learning*, foram utilizados os valores de $\alpha = 0,5$ para a taxa de aprendizado e $\gamma = 0,6$ para o fator de desconto. Para a estratégia de exploração, utilizou-se $\tau = 1$.

4.4.4 Resultados de Simulação

A Fig. 4.6 mostra a convergência do algoritmo *Q-Learning* para o cenário de simulação proposto. A convergência foi medida observando-se as modificações na tabela de valores Q durante a fase de aprendizado do algoritmo. O valor mais próximo de zero indica que as iterações não provocavam mais mudanças nos valores da função estado-ação. Como é mostrado, o aprendizado da melhor política leva aproximadamente 30000 iterações, o que corresponde a um intervalo de tempo de 30 segundos, utilizando a estrutura de quadros de transmissão do padrão LTE, já descrita anteriormente.

Embora pareça um valor elevado para o número de iterações, especialmente considerando a necessidade de operação em tempo real, algumas observações devem ser feitas. Em primeiro lugar, a solução não depende *a priori* do valor de *gap* de capacidade, como ocorre com o algoritmo de Levin-Campello. É necessário inicialmente apenas a solução do problema de AMC para o aprendizado dos limiares de razão sinal-ruído para a correta comutação entre as configurações de modulação e codificação. Em segundo lugar, diferentemente da abordagem de Levin-Campello, a solução que faz uso do aprendizado por reforço possui memória dos estados e, portanto, não precisa ser recalculada a cada novo início de transmissão, especialmente em ambientes de baixa variabilidade temporal. Parte das iterações pode ser realizada *off-line*, aproveitando os valores Q aprendidos durante a etapa inicial de convergência do algoritmo, até que se chegue a um estado terminal, isto é, uma configuração na qual não é possível tomar outra ação senão manter a alocação de *bits* atual.

Deve-se ainda considerar que os 30 segundos necessários para a convergência da estratégia, dependendo da aplicação do usuário, não representam um intervalo de tempo excessivamente longo. Por exemplo, de acordo com [112], sessões de tráfego *peer to peer* possuem duração média de 1800 segundos, transferências de fluxo FTP (*file transfer protocol*) possuem duração média de 180 segundos e navegação *web* e leitura de *email* possuem sessões de 60 segundos, em média. Logo, aplicações que dependem de taxas maiores de transmissão e implicam em maior tempo de sessão são capazes de tirar maior proveito da estratégia de *bit loading*.

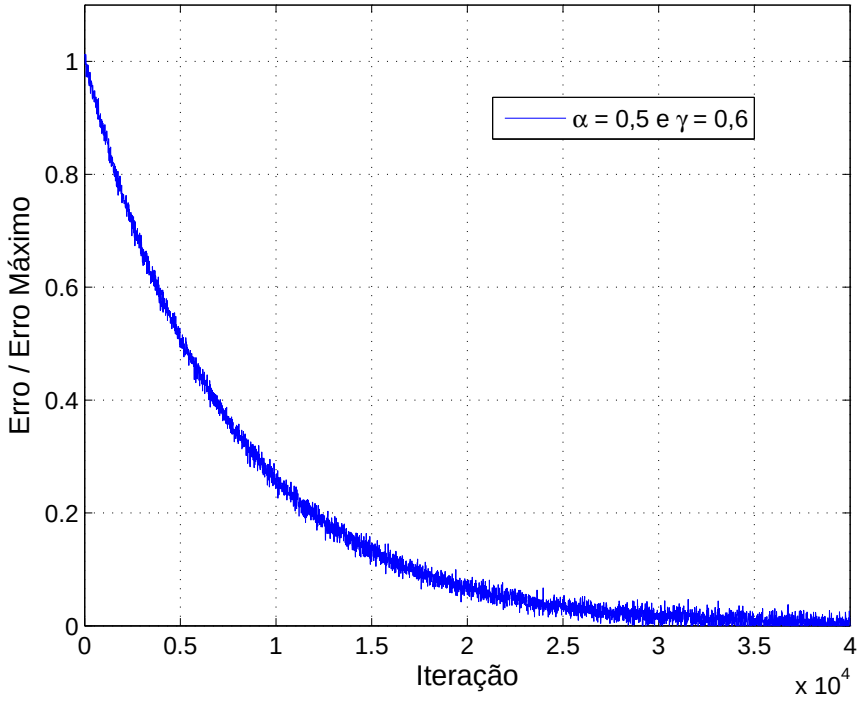


Figura 4.6: Curva de aprendizado do algoritmo *Q-Learning* para o problema de *bit loading* discreto.

A Fig. 4.7 mostra os resultados da taxa de erro de *bit* e da taxa de erro de pacote em função da razão E_b/N_o (a energia média de *bit* e a densidade espectral de potência unilateral do ruído) tomada entre todos os *resource blocks* para um cenário em que não é utilizada a estratégia de *bit loading*. Neste, a potência é alocada uniformemente entre os *resource blocks*, e a modulação é a mesma em todos os *resource blocks* utilizados. A modulação 4QAM com taxa de codificação $1/2$ se mostra a mais robusta, pois demanda menor quantidade de energia para manter uma dada BER do sistema. Por outro lado, modulações de ordem mais alta exigem maior energia para manter um dado limiar de erro de *bit* do sistema.

A Fig. 4.8 mostra, para este mesmo cenário, a eficiência espectral de cada um dos esquemas de modulação e codificação adaptativas utilizados. Como pode-se constatar, modulações de ordem mais baixa e com taxa de codificação menor, apesar de serem mais robustas à influência do ruído e do ganho de canal, apresentam menor eficiência espectral. Para valores suficientemente altos de

E_b/N_o, todas são capazes de atingir os limites de eficiência espectral dados pela Tabela 4.1. Este resultado motiva a utilização de *bit loading*, com o objetivo de ajustar a quantidade de informação transportada de acordo com as condições do canal experimentadas por cada *resource block*.

A Fig. 4.9 mostra o desempenho do sistema quando passa-se a utilizar o *bit-loading* discreto. É considerada ainda a comparação de duas soluções: a proposta, que utiliza o aprendizado por reforço (denotada por RL), e a que utiliza o algoritmo de Levin-Campello (marcada como LC). Para ambas as soluções, observa-se ganho de razão sinal-ruído da ordem de 3dB quando comparada à modulação mais robusta. Nota-se ainda que a solução proposta, que utiliza aprendizado por reforço, é ligeiramente pior do que a solução ótima obtida por Levin-Campello. Entretanto, esta perda de desempenho é inferior a $0,5$ dB, e se deve à quantização do valor de energia ainda disponível no transmissor. Este valor, conforme mostrado na seção 4.4.2, é utilizado como descritor de estados do sistema.

Entretanto, deve-se destacar que o algoritmo de aprendizado por reforço é capaz de aprender uma política de alocação de *bits* por meio da interação direta com o ambiente, dependendo apenas da solução do problema de modulação e codificação adaptativas, tratado na Capítulo 3. Uma vez aprendida a melhor ação para cada estado, basta a seleção da política gulosa para encontrar a alocação que maximiza a capacidade do sistema, sem a necessidade de recalcular toda a tabela com os valores da função Q a cada novo conjunto de quadros de transmissão. O algoritmo de Levin-Campello, por outro lado, deve ser executado, necessariamente, a cada conjunto de *resource blocks*, e depende da estimação do valor de *gap* Γ no qual o sistema opera (para as simulações realizadas, foi fixado de tal forma a se obter uma taxa de erro de pacote de 10%), e cuja obtenção analítica é, em geral, muito complexa, ou computacionalmente onerosa, pois depende da simulação Monte Carlo do cenário em questão, e pode variar de ambiente para ambiente [102, 104].

Finalmente, a Fig. 4.10 mostra a eficiência espectral das abordagens por aprendizado por reforço (denotada ainda por RL), e a que utiliza o algoritmo de Levin-Campello (identificado como LC). É possível identificar a tendência do *bit loading* de acompanhar a envoltória das curvas mostradas na Fig. 4.8, trabalhando sempre com a maior eficiência espectral possível. Percebe-se ainda que existe diferença na eficiência espectral atingida pelas abordagens de aprendizado por reforço e do algoritmo de Levin-Campello. Este possui desempenho melhor mas, em termos práticos, ambas apresentam diferença desprezível na eficiência espectral alcançada.

4.5 CONCLUSÃO

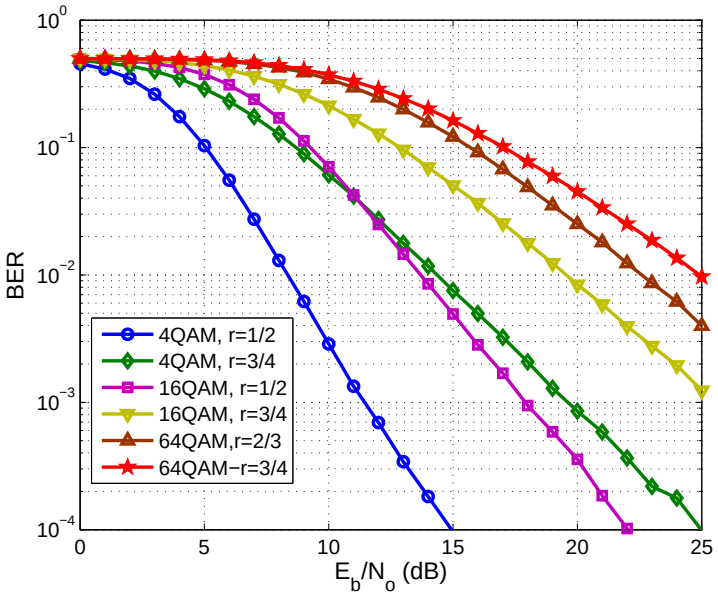
Neste capítulo, foi tratado o problema de alocação de *bits*, ou *bit loading*, em sistemas OFDM. Foi proposta uma solução baseada em aprendizado por reforço, utilizando o algoritmo *Q-Learning* para a obtenção de uma distribuição de *bits* entre os *resource blocks* de forma a maximizar a taxa

de transmissão do sistema.

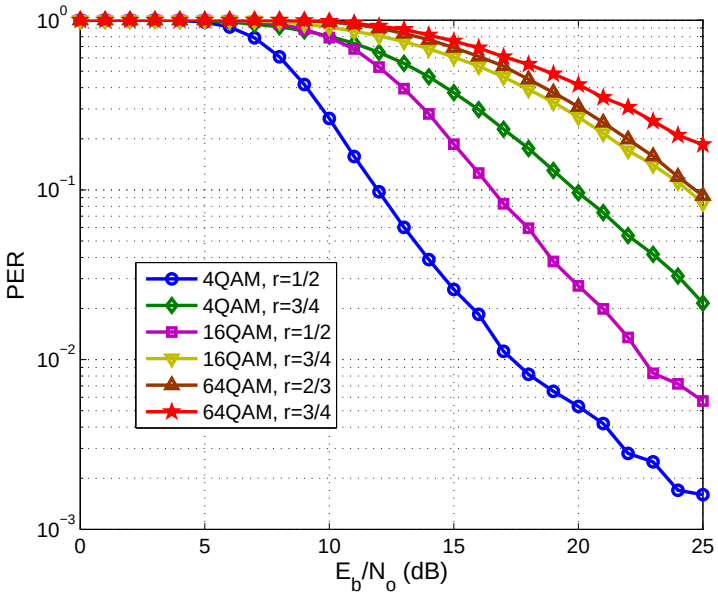
Por meio de resultados de simulação, a solução proposta foi comparada com a solução ótima, dada pelo algoritmo de Levin-Campello. Mostrou-se que a estratégia de aprendizado por reforço é capaz de se aproximar da solução ótima, porém sem a necessidade de tratamento analítico ou simulações computacionais realizadas *a priori* para a obtenção do valor de *gap* utilizado na formulação do algoritmo ótimo. O aprendizado por reforço depende apenas da solução prévia do problema de modulação e codificação adaptativas, que pode ser realizado de forma *on-line* de acordo com a estratégia proposta no Capítulo 3.

Verificou-se que, em termos de taxa de erro de *bit*, ambas estratégias foram capazes de fornecer ganhos da ordem de 3dB quando comparadas ao cenário em que nenhuma adaptação de enlace é utilizada. O algoritmo de Levin-Campello possui desempenho ligeiramente superior à solução via aprendizado por reforço, a aproximadamente $0,5$dB. Esta diferença deve-se ao descritor de estados utilizado, que utiliza informação discretizada a respeito da potência disponível para transmissão. Quando comparadas em termos de sua eficiência espectral, as soluções apresentam diferença desprezível, inferior a $0,1$bps/Hz.

O próximo capítulo traz a proposta de continuidade deste trabalho, que trata do problema de escalonamento em sistemas multi-usuário.



(a) Taxa de erro de bit



(b) Taxa de erro de pacote

Figura 4.7: Taxa de erro de *bit* e taxa de erro de pacote para o cenário em que é utilizada alocação de potência uniforme e a mesma modulação entre os *resource blocks* transmitidos.

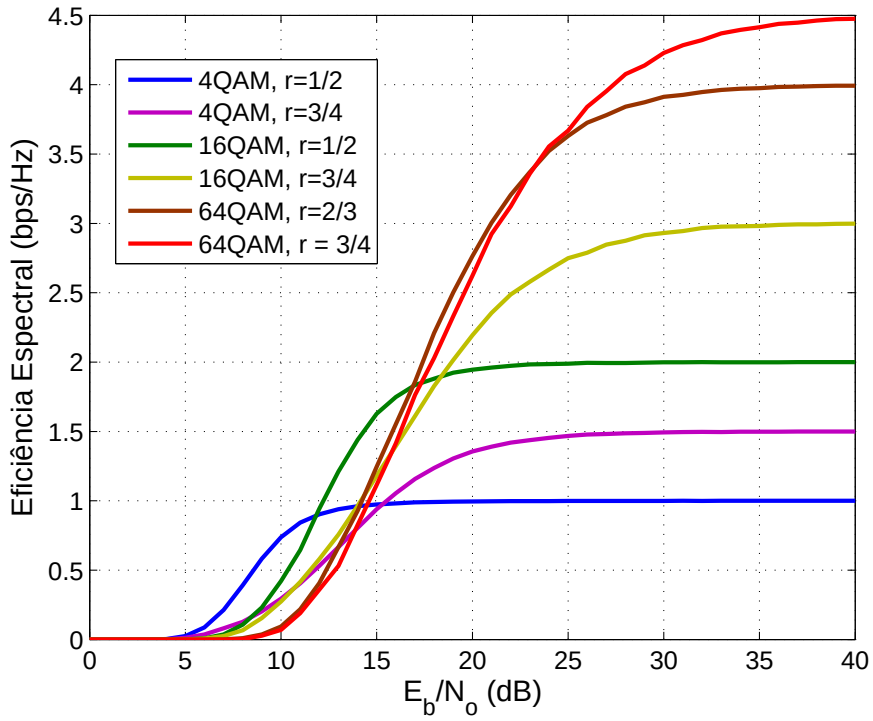
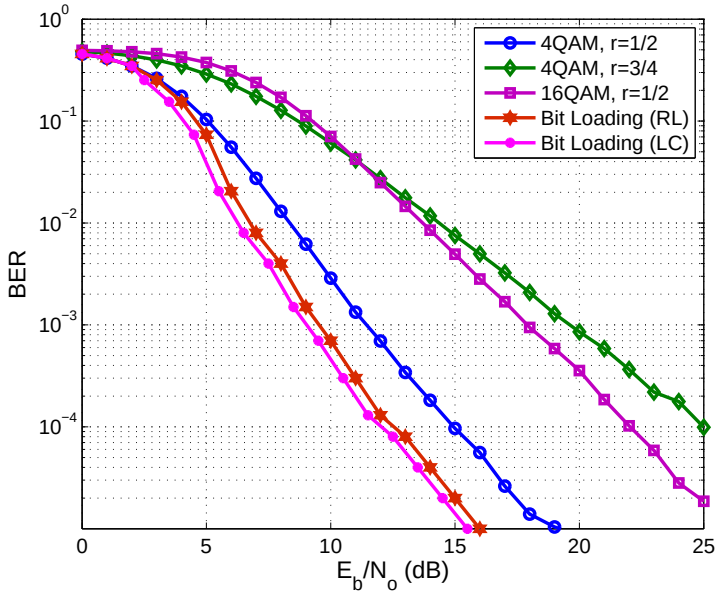
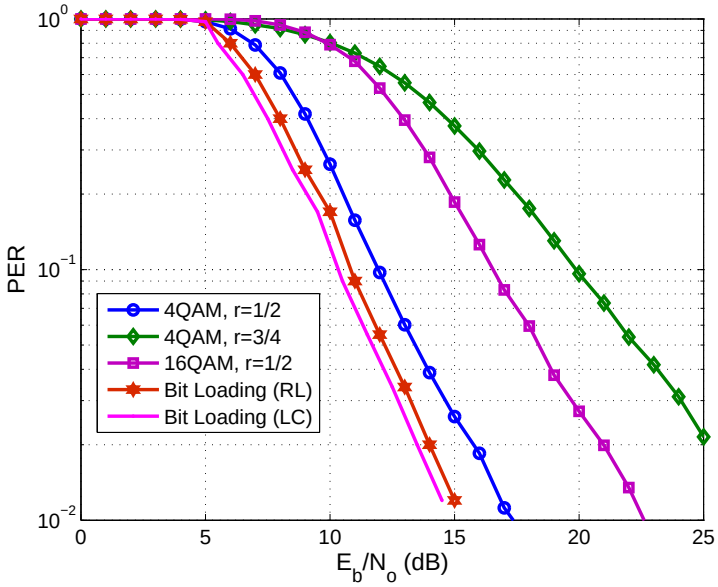


Figura 4.8: Eficiência espectral dos diferentes esquemas de modulação e codificação para o cenário em que é utilizada alocação de potência uniforme e a mesma modulação entre os *resource blocks* transmitidos.



(a) Taxa de erro de bit



(b) Taxa de erro de pacote

Figura 4.9: Taxa de erro de *bit* e taxa de erro de pacote para o cenário em que é utilizado *bit loading* discreto. É mostrado o desempenho das soluções por aprendizado por reforço (RL) e pelo algoritmo de Levin-Campello (LC).

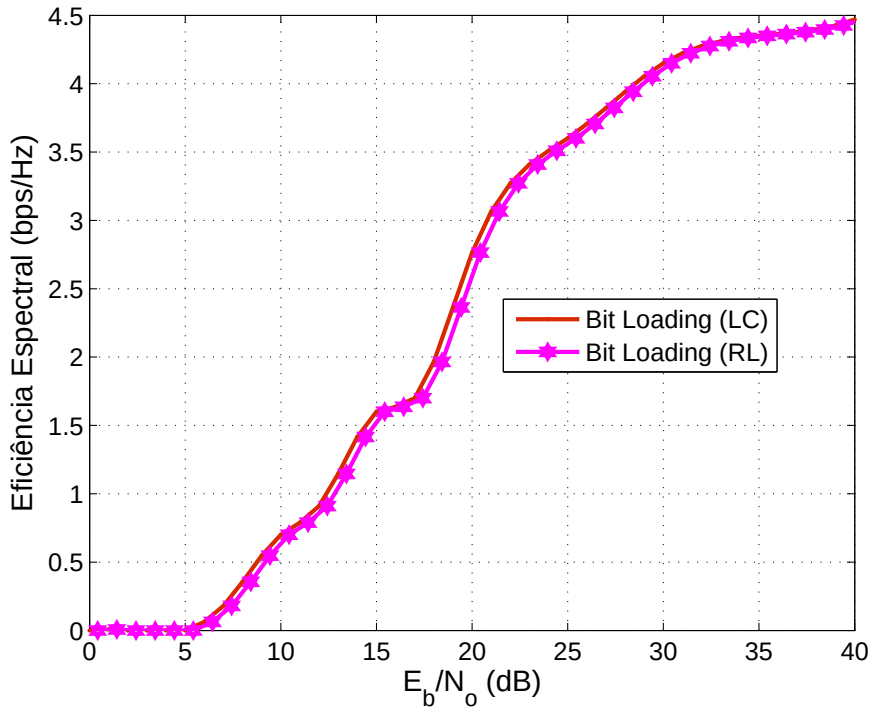


Figura 4.10: Eficiência espectral para a situação em que é utilizado *bit loading* discreto. É mostrado o desempenho das soluções por aprendizado por reforço (RL) e pelo algoritmo de Levin-Campello (LC).

5 FRAMEWORK PARA ALOCAÇÃO DE RECURSOS EM SISTEMAS ODFMA DE ALTA MOBILIDADE UTILIZANDO APRENDIZADO POR REFORÇO

5.1 INTRODUÇÃO

Conforme abordado nos capítulos anteriores, operar sistemas de comunicação com parâmetros fixos de transmissão não é conveniente, pois isso acaba por reduzir a vazão do sistema, uma vez que esta depende da relação existente entre o ganho do canal, tipo de ruído aditivo, modulação e codificação utilizadas, além da potência alocada. Por meio de estratégias como o AMC, abordada no Capítulo 3, e o *bit loading* discreto, tratado no Capítulo 4, é possível aproveitar melhor os recursos de transmissão, de forma a diminuir o hiato existente entre as capacidades teória e observadas em sistemas de comunicação OFDM.

Entretanto, a maximização da taxa de transmissão não é o único parâmetro a ser observado no projeto e operação dos sistemas digitais sem fio. Ainda que a transmissão da informação utilizando taxas de dados as mais altas possíveis vigore entre os principais objetivos de sistemas de comunicação celulares, outros objetivos devem ser atendidos.

Dentro do contexto de gerência de recursos de rádio, o escalonamento de usuários tem atraído considerável atenção, especialmente para os sistemas de quarta geração. O escalonador atua na estação rádio base e possui como função atribuir aos usuários os recursos de transmissão, como faixas do espectro e potência de transmissão, seguindo alguma política previamente estabelecida, de forma que todos possam obter acesso ao meio de comunicação para transmissão da informação [39]. Seu papel tem ganhado destaque devido à heterogeneidade das aplicações que trafegam nas redes celulares, como VoIP (voz sobre IP), transmissão de vídeo em tempo real e navegação *web*. Estas aplicações possuem diferentes requisitos de atraso de transmissão, largura de banda e perda de pacotes tolerada, o que gera desafios específicos para os algoritmos tradicionalmente utilizados para o escalonamento e seleção de usuários em redes celulares.

Tendo em vista este novo conjunto de restrições que se impõe para os ambientes de comunicação multi-usuário, este capítulo possui como objetivo apresentar uma proposta de estratégia de alocação de recursos e escalonamento de usuários utilizando técnicas de aprendizado por reforço multi-objetivo.

Em primeiro lugar, é feita uma breve revisão sobre a natureza dos atuais algoritmos de escalonamento. Em seguida, tratam-se os requisitos importantes e esperados para algoritmos de escalonamento para sistemas de quarta e próximas gerações. É apresentada a principal contribuição desse capítulo, que é a modelagem matemática do problema de alocação sob a ótica de algoritmos que se utilizam de aprendizado por reforço. Finalmente, serão apresentados os resultados de simulação para a avaliação do desempenho do *framework* apresentado. Para tal, será utilizado como cenário de simulação as características de transmissão presentes no enlace direto do padrão 3GPP-LTE [40], e será realizada a comparação do desempenho do algoritmo proposto com algumas estratégias clássicas encontradas na literatura.

5.2 ESTRATÉGIAS DE ESCALONAMENTO E ALOCAÇÃO

Nesta seção serão descritos, de forma breve, os principais algoritmos de escalonamento encontrados na literatura. Eles não estão restritos a nenhuma tecnologia específica de padrão ou sistemas de transmissão, e servem como referência para a proposta de novos algoritmos.

As estratégias usuais de escalonamento podem ser agrupadas de acordo com os objetivos a serem atendidos e seus parâmetros de entrada. São divididas em três conjuntos:

- Algoritmos que exploram informações da aplicação, ou *application aware*: estes foram os primeiros a serem utilizados, e têm como base algoritmos inicialmente utilizados para aplicações no campo de ciência da computação. São exemplos o algoritmo *first-in, first-out* (FIFO), que escalona os usuários de acordo com o instante de requisição de serviço, e *round-robin* (RR), que organiza os usuários em uma fila circular e distribui os recursos de escalonamento entre os usuário de modo sequencial, um a um, seguindo a ordem dessa fila [113]. Apesar de serem capazes de prover justiça na alocação de recursos, estes não exploram características do canal de comunicação visando maximizar a vazão da célula ou atender a requisitos de qualidade de serviço das diferentes aplicações.

- Algoritmos que exploram informações do canal de comunicação, ou *channel aware*: algoritmos dessa classe tendem a ser oportunistas, pois priorizam os usuários que possuem melhor qualidade de canal para a transmissão. Um dos exemplos que pode ser citado é o *maximum throughput* (MT), ou máxima taxa [113], que é capaz de atingir a maior eficiência espectral possível de uma célula, mas gera problemas de justiça na distribuição dos recursos de transmissão. Para tentar contorná-los, algoritmos como o *proportional fair* (PF) [114] foram propostos, mas não levam em consideração o comportamento do tráfego de cada aplicação, não sendo capazes de garantir qualidade de serviço [115].

- Algoritmos que exploram informações da aplicação e do canal de comunicação, ou *channel and application aware*: algoritmos desse grupo buscam coletar informações tanto da

aplicação quanto da resposta de canal de cada usuário, de forma a balancear vazão, justiça e qualidade de serviço, esta medida em termos do atraso máximo suportado pela aplicação ou da perda de pacotes. Como exemplo, pode-se citar o algoritmo M-LWDF (*modified largest weight delay first*) [39].

Como é possível constatar, os algoritmos utilizados são especificamente projetados para utilizar de forma ótima (em termos de vazão) os recursos espectrais ou promover justiça na alocação dos recursos de rádio entre os usuários. Além disso, é difícil demonstrar a capacidade desses algoritmos em satisfazer restrições de qualidade de serviço, como atrasso máximo no tempo de entrega de pacotes ou taxa de perda de pacotes [39].

5.3 REQUISITOS PARA ALGORITMOS DE ESCALONAMENTO

Com base no exposto na seção 5.2, e lembrando da heterogeneidade de aplicações que atualmente trafegam em redes celulares, é possível enumerar algumas características que um algoritmo de alocação de recursos e de escalonamento de usuários deveria possuir [39]:

- Baixa complexidade e escalabilidade: nos sistemas atuais (e, de forma mais específica, o LTE), o escalonador de pacotes trabalha com uma granularidade temporal da ordem de 1ms, ou seja, as decisões de escalonamento de usuários devem ser tomadas a cada TTI. Encontrar a melhor forma de alocação de recursos por meio de técnicas de otimização não linear ou busca exaustiva [116, 117, 118] torna-se proibitivo em termos de custo computacional e tempo de processamento.
- Eficiência espectral: a utilização eficiente do espectro eletromagnético está entre as principais metas de sistemas de comunicações sem fio. Para tal, busca-se explorar a diversidade multiusuário, selecionando e priorizando usuários que apresentam a melhor resposta de canal.
- Justiça: apesar de estratégias que visam a maximização de taxa de transmissão serem capazes de prover a melhor utilização do canal de comunicação em termos de eficiência espectral, estas também tendem a distribuir de forma desigual os recursos do sistema de comunicação, penalizando usuários que possuem canais de comunicação mais seletivos ou que se encontram próximos à borda da célula. É necessário que esses usuários também possuam oportunidades de transmissão e utilização dos recursos celulares.
- Qualidade de Serviço (QoS, *quality of service*): atender aos requisitos de QoS das diferentes aplicações está entre as principais funções dos algoritmos de escalonamento, sobretudo nos sistemas de comunicação de quarta geração. De forma geral, estes requisitos variam de

acordo com o tipo de serviço (tráfego de vídeo em tempo real, voz sobre IP, navegação *web* etc.) e são especificados em termos de taxa mínima de transmissão, atraso de pacote e probabilidade de perda de pacote.

Pode-se ainda acrescentar à lista outro requisito desejável, que é a capacidade de tomar decisões em um horizonte maior de tempo. Todos os algoritmos citados apresentam, de certa forma, comportamento míope com respeito a esse requisito, visto que o horizonte de tomada de decisão para a seleção de usuários é imediato: nenhuma consideração a respeito de mudanças no estado do canal ou no tráfego gerado é feita ao se escolher quais usuários terão direito de transmissão, o que pode implicar subaproveitar recursos de transmissão. Naturalmente, considerar um horizonte maior de tomada de decisão de escalonamento implica que o algoritmo deve ser capaz de prever, de alguma forma, o comportamento a longo prazo da resposta de canal ou das características de tráfego de cada usuário.

5.4 PROPOSTA DE ESTRATÉGIA DE ESCALONAMENTO

A existência de diferentes objetivos de transmissão, estes muitas vezes concorrentes e conflitantes (como citado na seção 5.3), sugere que o escalonamento de usuários pode ser tratado como um problema de otimização multi-objetivo. Tendo em vista as diferentes restrições e tipos de tráfego que podem coexistir nas redes celulares atuais e futuras, busca-se uma solução de compromisso entre atraso de transmissão, vazão, taxa de perda de pacote e justiça na alocação dos recursos. Além disso, pode ainda ser desejável a utilização de estratégias que sejam capazes de rastrear as variações do canal de comunicação e dos requisitos de cada usuário. Técnicas de aprendizado de máquina mostram-se, portanto, sugestivas para promover este tipo de aprendizado e adaptação.

Dessa forma, faz-se a proposta de uma estratégia de seleção de usuários e alocação de recursos que possui como base o aprendizado por reforço em sua versão multi-objetivo, conforme detalhado no Capítulo 2. Um diagrama de blocos simplificado, ilustrando as estratégias de operação desse sistema, é mostrado na Fig. 5.1.

Em primeiro lugar, é necessária uma representação apropriada para o problema, de forma a resolvê-lo por meio das estratégias de aprendizado por reforço. A representação deve ser tal que permita encontrar políticas de escalonamento que ofereçam soluções de compromisso entre os diferentes requisitos de QoS. Por meio da representação dada pela Fig. 5.1, é interessante que o agente inteligente monitore e utilize como descritor de estados o nível de preenchimento do *buffer* e o atraso experimentado por cada pacote neste mesmo *buffer*. Dessa forma, espera-se ser possível contemplar os diferentes requisitos de perda de pacote, taxa de transmissão e atraso máximo de escalonamento.

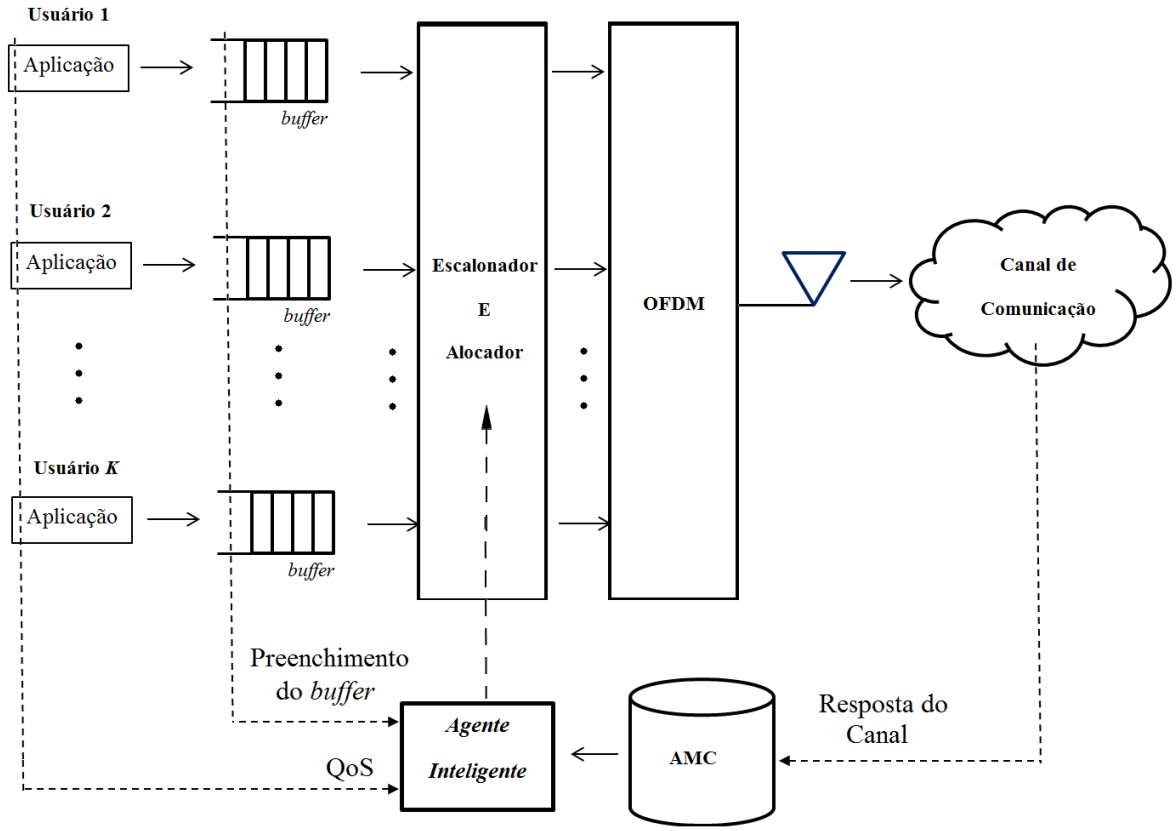


Figura 5.1: Ilustração da operação de escalonamento utilizando um agente de aprendizado por reforço.

É ainda necessária a modelização das ações disponíveis ao agente, que consiste em determinar qual o melhor *resource block* e o melhor instante de alocação de cada usuário, tomando como referência a duração de um quadro LTE, ou seja, um horizonte de 10 ms [119, 120]. Requer-se ainda a obtenção de um sinal de recompensa (reforço) que traduza os requisitos de qualidade de serviço de cada aplicação para cada usuário. Conforme mostrado na seção 5.3, esses requisitos são medidos, geralmente, em termos da taxa de transmissão, atraso de escalonamento e justiça no serviço dos usuários.

Nas próximas subseções, serão apresentadas as soluções propostas para implementar os requisitos supracitados.

5.4.1 Proposta de Representação de Estados, Ações e Recompensas

A otimização do problema de escalonamento multiusuário demanda analisar a situação de todos os usuários ao mesmo tempo, o que envolveria uma complexidade computacional elevada para o problema em questão, pois seriam necessários descritores de estados que exigiriam uma quantidade de estados a serem representados da ordem de $\mathcal{O}\left(\upsilon^{N_{usuarios}}\right)$, em que υ é uma

constante que depende do números de descritores utilizados para a representação dos estados, e $N_{usuarios}$ é o número de usuários presentes na célula, para os quais se deseja alocar os recursos de transmissão.

Dessa forma, em vez de tratar o problema considerando um espaço de representações tomando a presença de todos os usuários, a modelagem do problema por aprendizado por reforço levará em consideração a representação de estados, ações e recompensas por usuário, de forma que a complexidade da representação por usuário será de $\mathcal{O}(\upsilon)$, e a complexidade de solução do problema será de $\mathcal{O}(\upsilon N_{usuarios})$. Apesar de não consistir na abordagem ótima, há considerável redução tanto da complexidade computacional quanto na dimensionalidade da representação do problema, tornando-o tratável sob a ótica do aprendizado por reforço.

Deve-se ainda lembrar que, em um dado instante t, deve ser tomada uma decisão de escalonamento. Uma nova decisão deve ser tomada após $t + \Delta t$ segundos, em que Δt representa o intervalo de decisão de alocação e transmissão (ou TTI que, para o caso do LTE, vale 1 ms). Dessa forma, para um determinado usuário, os estados são descritos pela tripla:

$$s_t = \begin{bmatrix} \rho_{t-1} & HOL_t & B_t \end{bmatrix} \tag{5.1}$$

em que ρ_{t-1} representa a eficiência espectral alcançada pela última decisão de escalonamento, HOL_t representa o atraso do primeiro pacote que se encontra no *buffer* de transmissão do usuário no instante t, e B_t representa, em porcentagem, o nível de ocupação do *buffer* de transmissão do usuário. Para evitar o problema de lidar com uma variável contínua, o valor B_t é quantizado em intervalos de 5%, isto é, $B_t \in \{0, 5\%, 10\%, \cdots, 100\%\}$. É interessante ainda observar que o valor de HOL_t já se encontra quantizado, uma vez que as decisões de escalonamento são tomadas a cada Δt segundos, sendo essa a unidade básica de tempo utilizada.

Uma ação é definida pela dupla:

$$a_{ij} = \begin{bmatrix} RB_i & t + j\Delta t \end{bmatrix} \tag{5.2}$$

que significa alocar no i-ésimo *resource block* do *slot* do instante de tempo $t + j\Delta t$ os pacotes a serem transmitidos para um determinado usuário. Naturalmente, $1 \leq i \leq N_{RB}$, em que N_{RB} representa o número máximo de *resource blocks* em um TTI (o que é uma função da banda disponível), e $0 \leq j \leq 9$, uma vez que o horizonte considerado para a estratégia de escalonamento proposta é de um quadro LTE de transmissão, ou seja, 10 TTI's. A Fig. 5.2 procura ilustrar os conceitos apresentados por esse procedimento.

A recompensa é definida de acordo com o tipo de serviço demandado pelo usuário. Por se tratar de um problema multidimensional, a recompensa é definida pelo vetor:

$$\vec{r}(s, a_{ij}) = \begin{bmatrix} r_1(s_t, a_{ij}) & r_2(s_t, a_{ij}) & r_3(s_t, a_{ij}) \end{bmatrix} \tag{5.3}$$

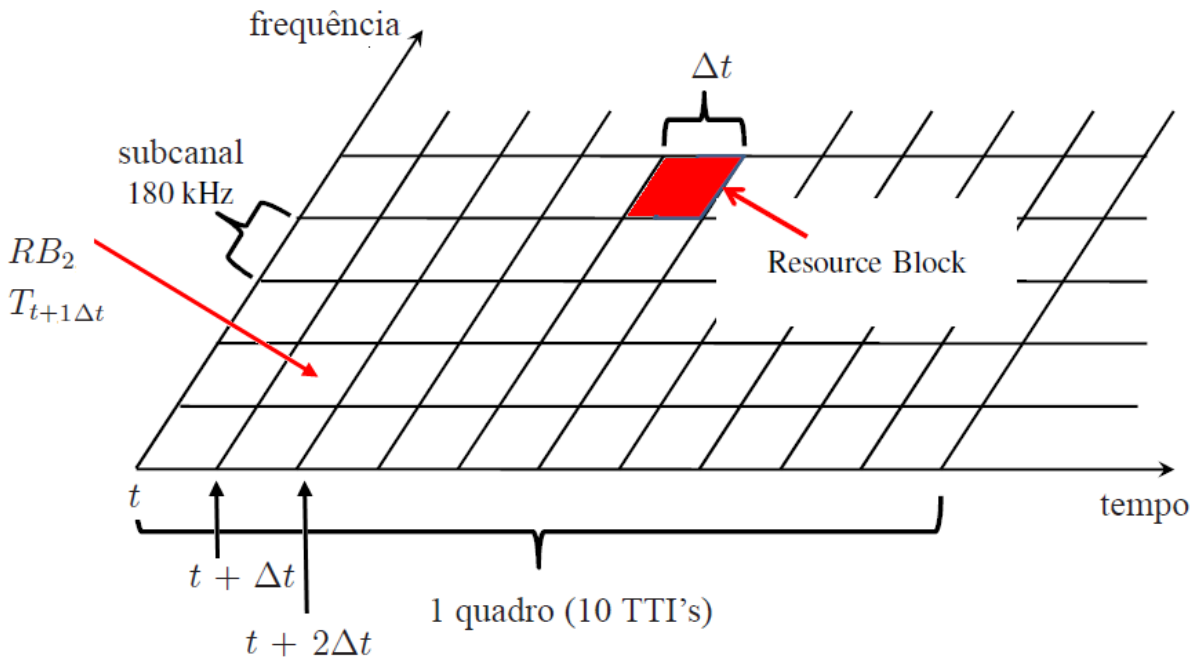


Figura 5.2: Ilustração da estrutura do quadro de transmissão e as ações possíveis.

em que $r_1(s_t, a_{ij})$ $r_2(s_t, a_{ij})$ e $r_3(s_t, a_{ij})$ referem-se às três dimensões de recompensa analisadas pelo *framework* proposto, que serão detalhadas a seguir.

A recompensa $r_1(s_t, a_{ij})$ representa a eficiência espectral normalizada da ação tomada. Esta tese propõe a seguinte forma de cálculo da recompensa:

$$r_1(s_t, a_{ij}) = \frac{\rho\left(a_{ij}\right)}{\max\limits_{k} \rho\left(MCS_k\right)} \tag{5.4}$$

em que $\rho\left(a_{ij}\right)$ representa a eficiência espectral máxima que pode ser atingida ao se transmitir utilizando o i-ésimo *resource block* do *slot* localizado no instante de tempo $j\Delta t$, supondo a alocação de potência uniforme entre todos os *resource blocks*, e $\rho\left(MCS_k\right)$ representa a máxima eficiência espectral disponível ao sistema (isto é, representa a eficiência espectral do esquema de modulação e codificação que possui modulação de ordem mais alta e maior taxa de codificação). Dessa forma, a recompensa é normalizada para que seu valor máximo seja 1 (a importância desse fato ficará clara nos próximos parágrafos).

A recompensa $r_2(s_t, a_{ij})$ deve refletir o tempo de entrega de um pacote, devendo ser, portanto, inversamente proporcional ao atraso experimentado pelo pacote, de forma a priorizar ações que não atrasem a entrega dos dados, ou deixem o pacote por muito tempo na fila do *buffer* de transmissão. Propõe-se utilizar a forma:

$$r_2(s_t, a_{ij}) = \begin{cases} 1, & \text{se } HOL_{t+j\Delta t} \leq T_{limite} \\ e^{-\beta_1\left(HOL_{t+j\Delta t} - T_{limite}\right)}, & \text{caso contrário} \end{cases} \tag{5.5}$$

em que T_{limite} representa o limite tolerável de atraso para a aplicação em questão (que depende dos requisitos de QoS da mesma), e β_1 é uma constante positiva, escolhida pelo projetista do sistema de forma a penalizar atrasos excessivos no escalonamento de um conjunto de pacotes.

Cabe notar ainda que essa definição de recompensa apenas possui sentido para aplicações que são sensíveis ao atraso. Para o caso em que a aplicação não é sensível ao atraso, é mais interessante definir:

$$r_2(s_t, a_{ij}) = K_{r_2} \tag{5.6}$$

em que $K_{r_2} < 1$ é uma constante, de forma a priorizar o escalonamento das aplicações que são sensíveis ao atraso ou sujeitas a um limite máximo de tempo para a entrega de pacotes.

A recompensa $r_3(s_t, a_{ij})$ tem como objetivo controlar o nível de ocupação do *buffer* de transmissão de cada usuário, sendo inversamente proporcional à porcentagem de ocupação desse mesmo *buffer*. Ela é definida por meio da relação

$$r_3(s_t, a_{ij}) = \begin{cases} 1, & \text{se } B_{t+j\Delta t} \leq B_{limite} \\ e^{-\beta_2\left(B_{t+j\Delta t} - B_{limite}\right)}, & \text{caso contrário} \end{cases} \tag{5.7}$$

em que B_{limite} representa um valor limite aceitável para o nível de preenchimento do *buffer* de transmissão do usuário, valor esse que depende da QoS que se deseja garantir aos usuários de acordo com suas aplicações, sendo também uma escolha do projetista do sistema.

5.4.2 Solução do Problema de Aprendizado

Uma vez determinados os valores de recompensa para cada um dos usuários e cada uma das ações que é possível de ser tomada, deve-se buscar as soluções não dominadas para cada um dos usuários, independentemente da política que é seguida. Para tal, é utilizado o algoritmo de aprendizado por reforço multi-objetivo com iteração no fecho convexo, conforme descrito no Capítulo 2, cuja iteração é repetida abaixo para a conveniência do leitor:

$$\overset{o}{Q}(s,a) \leftarrow \mathbb{E}\left\{ \vec{r}(s,a) + \gamma \text{hull} \bigcup_{a'} \overset{o}{Q}(s',a') \,|(s,a) \right\} \tag{5.8}$$

Naturalmente, para promover sua operação em tempo real, o operador média $\mathbb{E}$ foi substituído por uma iteração de valores baseada no método das diferenças temporais, conforme descrito também no Capítulo 2.

Após determinado o conjunto de ações não dominadas, representadas por $\overset{o}{A}$, faz-se necessário ranqueá-las. Para tal, é determinado o valor da recompensa resultade $R(s_t, a_{ij})$, com $a_{ij} \in \overset{o}{A}$, obtido por meio da combinação linear das recompensas nas diferentes dimensões, conforme sugere [53]:

$$\begin{aligned} R(s_t, a_{ij}) &= r_\alpha(s_t, a_{ij}) \\ &= \vec{\alpha} \cdot \vec{r}(s_t, a_{ij}) \\ &= \alpha_1 r_1(s_t, a_{ij}) + \alpha_2 r_2(s_t, a_{ij}) + \alpha_3 r_3(s_t, a_{ij}) \end{aligned} \tag{5.9}$$

Deve-se então, para esse caso, buscar um conjunto de pesos α_1, α_2 e α_3 de forma a combinar as recompensas em cada uma das dimensões, e esta é uma decisão do projetista do sistema. Caso deseje-se privilegiar soluções que reduzam o atraso de escalonamento, seleciona-se um valor mais elevado para α_2. Caso se busquem soluções que maximizem a vazão do sistema, deve-se aumentar o valor de α_1. A única restrição a ser feita é garantir que $\alpha_1 + \alpha_2 + \alpha_3 = 1$.

É interessante ainda perceber que, como as recompensas em todas as dimensões encontram-se normalizadas para o valor máximo de 1, não há o risco de a recompensa de uma dimensão sobrepor-se a outra, mascarando a decisão de escalonamento.

5.4.3 Seleção e Escalonamento dos Usuários

Uma vez ranqueadas as soluções não dominadas, deve-se proceder para o próximo passo, que consiste na seleção dos usuários que possuem o direito de transmissão em um determinado *resource block*. Como o problema é resolvido individualmente para cada usuário, pode ocorrer de dois ou mais usuários apresentarem como solução de recompensa combinada mais alta a transmissão em um mesmo *resource block* no mesmo instante de tempo. Nessa situação, é necessário algum critério de desempate.

Como forma de tentar prover alguma justiça na alocação de rescursos, esta tese propõe como critério de desempate a utilização de uma matriz de oportunidades de transmissão, identificada por $\boldsymbol{TO}$. Essa matriz possui 10 colunas, referentes aos índice j utilizado para indexar os *slots* de transmissão, e N_{RB} linhas, referindo-se ao índide i utilizado para indexar o *resource block* de transmissão.

A matriz é preenchida da seguinte forma: cada elemento $\boldsymbol{TO}_{ij}$ contém os usuários que podem transmitir no i-ésimo *resource block* no instante $t + j\Delta t$ (um usuário possui direito de transmissão pela ação a_{ij} caso esta pertença ao seu fecho convexo, determinado na etapa anterior do processo de escalonamento).

Nesse caso, procede-se da seguinte forma:

1. Para a trasmissão no i-ésimo *resource block* no instante $t + j\Delta t$, seleciona-se o usuário que

pertence a $\boldsymbol{TO}_{ij}$ e que possui o maior valor de recompensa combinada $R(s_t, a_{ij})$.

2. Caso haja empate entre os usuários, é selecionado aquele que aparece o menor número de vezes na matriz $\boldsymbol{TO}$.

3. Caso permaneça o empate, é feito um sorteio, selecionando de forma aleatória um dos usuários.

Na próxima iteração do processo de seleção e escalonamento, faz-se $t \leftarrow t + \Delta t$, observam-se os novos estados para os usuários, e repetem-se as iterações desde a atualização do fecho convexo até a nova etapa de seleção e escalonamento, descrita nessa subseção. Um diagrama geral do procedimento de seleção e escalonamento está resumido e ilustrado na Fig. 5.3.

5.4.4 Estratégias de Predição

É interessante notar que, na definição das Eqs. (5.4) e (5.7), faz-se necessário o conhecimento dos valores $\rho\,(a_{ij})$ e $B_{t+j\Delta t}$. O primeiro diz respeito à eficiencia espectral no i-ésimo *resource block* do instante $t+j\Delta t$, que dependerá do ganho do canal em $t+j\Delta t$ no conjunto de frequências ocupadas pelo *resource block* em análise. O segundo fator, $B_{t+j\Delta t}$, diz respeito à ocupação do *buffer* do usuário no instante $t + j\Delta t$.

Como optou-se por trabalhar com um horizonte de decisões de escalonamento superiores a um TTI (1 ms), faz-se necessário, ao estimar as recompensas, que se tenha conhecimento da resposta futura do canal e do nível de ocupação do *buffer* de transmissão de cada usuário para os instantes de tempo em questão. Para trabalhar com essa informação atualizada, são necessárias estratégias de predição de resposta de canal e de nível de ocupação de *buffer*, tal como ilustrado na Fig. 5.4.

Para realizar o processo de predição, esta tese tomou como referência os trabalhos [121, 122], que abordam estretégias de predição de canal baseadas em teoria de filtros adaptativos, em particular utilizando o algoritmo *set-membership affine projection* [123] para a atualização dos coeficientes do filtro adaptativo. A motivação para sua escolha reside no fato de a classe de algoritmos *set-membership affine projection* não apresentar solução de compromisso entre o tempo de convergência do algoritmo adaptativo e o erro de predição no estado estacionário, além de reduzida complexidade computacional quando comparado a outros algoritmos clássicos. Os detalhes quanto às propriedades de algoritmos adaptativos e à sua aplicação à predição de canal em sistemas OFDM podem ser encontrados nessas referências. Esta seção possui como objetivo apenas mostrar como a ideia original de predição de canal foi adaptada para realizar também a predição do nível de preenchimento de *buffer*.

De forma geral, quando filtros adaptativos são utilizados para realizar a predição, estes devem ser capazes de fornecer uma estimativa futura para o valor do sinal $s(n)$ que é apresentado em sua entrada. Logo, o valor de $s(n)$ é utilizado como entrada desejada $d(n)$ do sistema, e o valor

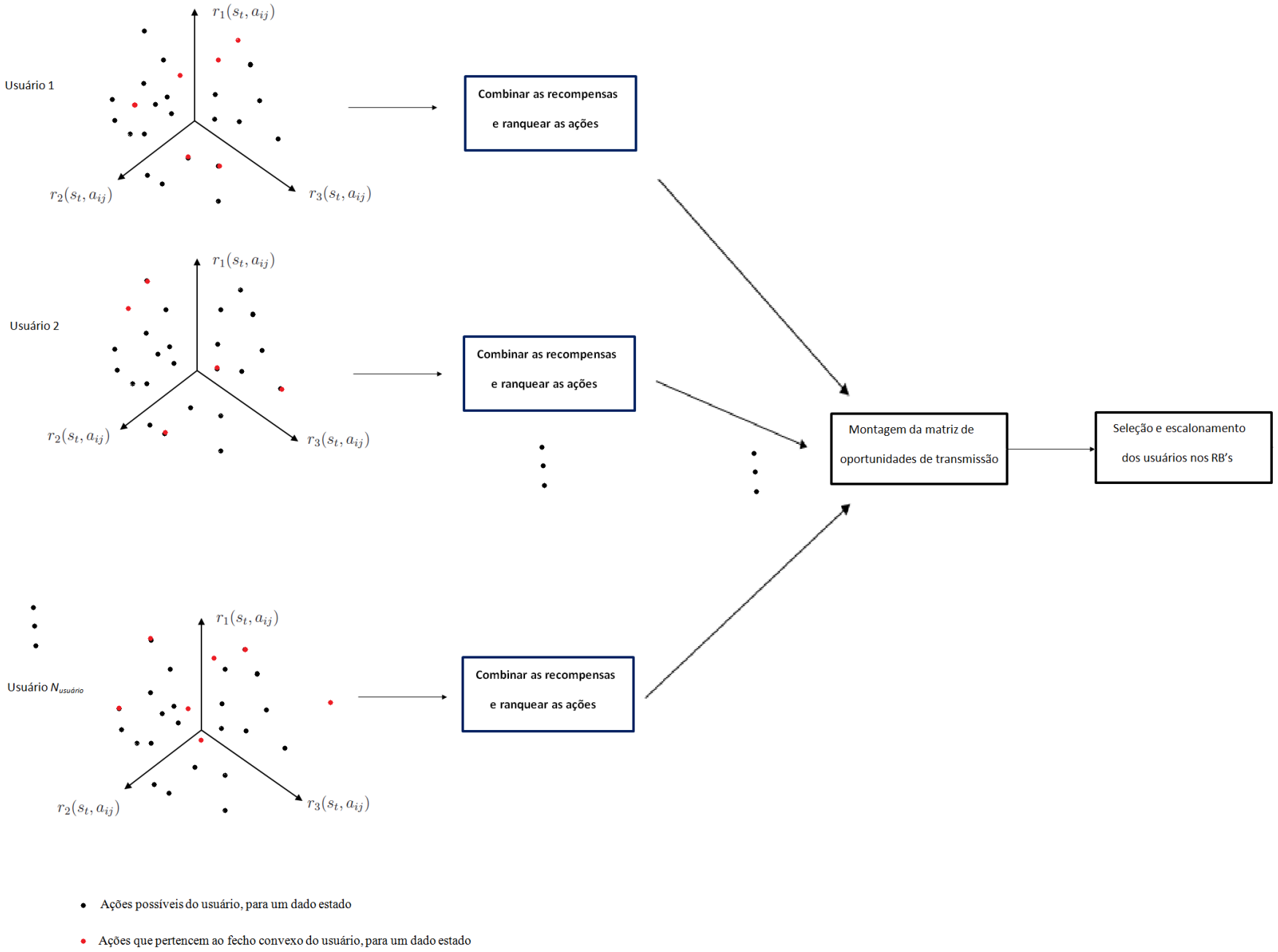


Figura 5.3: Resumo do procedimento de alocação de recursos para o ambiente multiusuário.

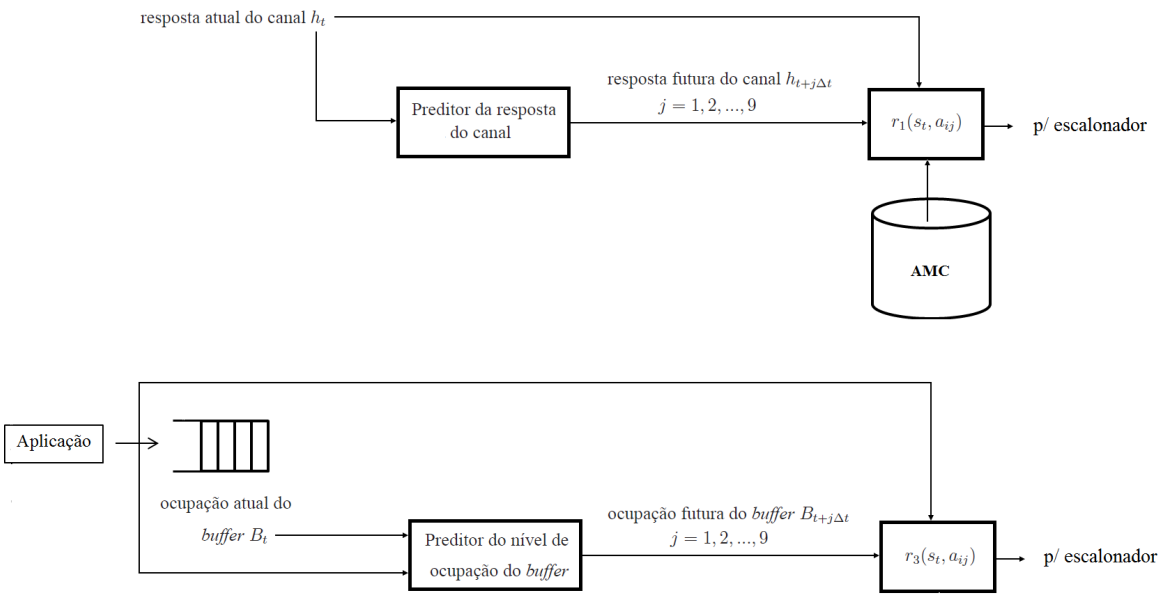


Figura 5.4: Ilustração da necessidade de estratégias de predição.

passado da entrada, denotado por $s(n-\ell)$, é utilizado como entrada do filtro adaptativo, conforme mostrado na Fig. 5.5. O parâmetro ℓ é o horizonte de predição, e denota o número de instantes de tempo para o qual a resposta do sinal de entrada deve ser predita à frente do instante de tempo n. Naturalmente, supõe-se que a saída do filtro está relacionada com sua entrada por meio da relação:

$$\begin{aligned} r(n) &= \sum_{i=0}^{N_c-1} c_i(n)s(n-i) \\ &= \mathbf{c}^H(n)\mathbf{s}(n), \end{aligned} \tag{5.10}$$

em que c_0, c_1, ..., c_{N_c-1} são os N_c coeficientes do filtro, que representam sua resposta impulsional. Como esta resposta possui comprimento finito, o filtro é dito do tipo FIR (do inglês, *finite impulse response*). Assim, cabe ao algoritmo adaptativo buscar o conjunto de valores dos coeficientes c_0, c_1, ..., c_{N_c-1} que minimizam o sinal de erro $e(n)$, definido como $e(n) = d(n) - r(n)$, que é a diferença entre o sinal que se deseja prever e a saída do filtro adaptativo. O índice n deve ser entendido como o instante t para o qual se realiza a predição.

Para a aplicação de predição do nível de ocupação do *buffer*, tomando como base a Fig. 5.5, utilizou-se como sinal de entrada $s(n)$ o número de *bytes* que são gerados pela aplicação no instante t. O horizonte de predição ℓ, para efeitos de simplificação, foi tomado como $\ell = 1$ pois, nesse caso, a atualização para os valores futuros de ocupação do *buffer* é feita por meio de um simples ajuste linear (confome será exposto na próxima equação), supondo que o algoritmo adaptativo já opera em convergência para os valores dos coeficientes do filtro. Dessa forma,

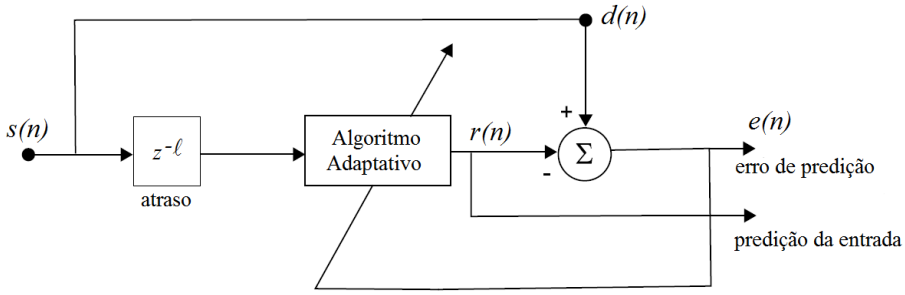


Figura 5.5: Estrutura adaptativa aplicada à predição do comportamento do sinal de entrada.

$r(n)$ retorna o número de bytes que serão gerados no próximo instante de tempo. Naturalmente, conhecendo-se o número de pacotes que serão gerados pela aplicação, o novo valor para o preenchimento do *buffer* pode ser calculado de acordo com

$$B_{t+j\Delta t} = B_t + j\left(\frac{r(n)}{B_{max}}\right) \cdot 100 \tag{5.11}$$

em que B_t é a ocupação atual do *buffer*, em porcentagem, e B_{max} é a capacidade máxima do buffer, medida em bytes. O valor obtido deve ainda ser quantizado em um dos valores permitidos para a modelização dos estados do sistema, conforme exposto em 5.4.1.

Utilizando o algoritmo adaptativo *set-membership affine projection*, os coeficientes do filtro adaptativo representado pela Eq. (5.10) são atualizados de acordo com a relação [123]:

$$\mathbf{c}(n+1) = \mathbf{c}(n) + \mathbf{S}(n)\left[\mathbf{S}^H(n)\mathbf{S}(n)\right]^{-1} e(n)\kappa(n)\mathbf{v}_1, \tag{5.12}$$

em que

$$\kappa(n) = \begin{cases} 1 - \frac{\nu}{|e(n)|}, & \text{se } |e(n)| > \nu \\ 0, & \text{caso contrário} \end{cases} \tag{5.13}$$

em que ν representa o valor limite para o erro de predição,

$$\mathbf{v}_1 = [1\ 0\ \ldots\ 0]^T \tag{5.14}$$

e

$$\mathbf{S}(n) \triangleq \Big[\mathbf{s}(n) \quad \mathbf{s}(n-1) \quad \ldots \quad \mathbf{s}(n-N_P+1)\Big] \tag{5.15}$$

é uma matriz que contém os últimos N_P vetores de regressores utilizados na Eq. (5.10). De forma simplificada, o valor de N_P é uma constante que determina qual deve ser a influência das amostras passadas (memória) na atualização dos coeficientes do filtro.

5.5 AVALIAÇÃO DO DESEMPENHO DA ESTRATÉGIA

Conforme tratado nas seções anteriores desse capítulo, é desejável que um algoritmo de escalonamento busque não apenas maximizar a eficiência espectral do sistema de comunicação, mas procure atender, em uma solução de compromisso, aos requisitos de qualidade de serviço de diferentes tipos de aplicação, que não estão limitados apenas à taxa de transmissão. Tendo em vista essa restrição, realizaram-se simulações para três tipos de aplicação: transmissão de vídeo tempo real, VoIP e navegação *web*. As duas primeiras aplicações são sensíveis ao atraso, possuindo restrição quanto ao tempo de escalonamento e entrega dos pacotes, diferentemente da terceira aplicação, servida de acordo com a regra *best effort*. Para verificar o comportamento da solução proposta, esta teve seu desempenho comparado com as abordagens clássicas da literatura, que são os algoritmos máxima taxa, *proportional fair* e M-LWDF, este considerado o mais apropriado para o escalonamento de usuários que possuem aplicações em tempo real com restrições de atraso de entrega de dados [124].

Inicialmente, serão apresentados o cenário de simulação e o ajuste de valores para os diferentes tipos de algoritmo de seleção e escalonamento de usuários. Em seguida, serão mostrados os resultados de simulação do desempenho dos algoritmos de escalonamento tanto em termos de taxa de transmissão alcançada quanto em termos de justiça na distribuição dos recursos e o atendimento de métricas de qualidade de serviço. Por último, são apresentadas as conclusões do capítulo.

5.5.1 Parâmetros de Simulação

Esta seção resume os parâmetros de simulação para ambiente multiusuário no qual os algoritmos de escalonamento foram validados e comparados. A Tabela 5.1 apresenta os parâmetros referentes ao canal de comunicação utilizado, implementado de acordo com as recomendações do modelo espacial do 3GPP [82, 83] e já utilizado nas simulações anteriores do trabalho. Destaca-se o fato de realizar as simulações em ambientes de baixa e de alta mobilidade.

A Tabelas 5.2 e 5.3 apresentam as configurações de transmissão utilizadas. Como forma de simplificar a tarefa de alocação de potência de transmissão, esta é distribuída uniformemente ao longo de todas as frequências utilizadas para transmissão, ou seja, a potência máxima da estação rádio base é distribuída de modo uniforme entre todos os *resource blocks* presentes em um intervalo de transmissão. Nesse caso, a tarefa de alocação é simplificada pois, a rigor, para otimizar a capacidade de transmissão, deve-se resolver simultânemante o problema de alocação de potência e seleção de usuários, já que existe a dependência entre os usuários para os quais é cedido o direito de transmissão e a potência que é alocada, pois esta é função da resposta do canal de cada usuário. Essa hipótese simplificadora não compromete severamente a vazão do sistema no enlace direto, haja vista que não há ganhos significativos, em termos de capacidade,

ao se realizar alocação de potência não uniforme no enlace direto, contrariamente ao que ocorre no enlace reverso [40, 125].

As características do tráfego gerado seguem as especificações descritas em [112]: em termos de taxa de transmissão, a transmissão de vídeo tempo real gera tráfego a uma taxa média de 1,65 Mbps; VoIP, a uma taxa de 64 Kbps; finalmente, a navegação *web* gera dados a uma taxa média de 200 Kbps. Outras especificações quanto à duração da sessão e duração dos bursts de dados são dadas também em [112]. Cabe ainda ressaltar que o prazo máximo para entrega de pacotes tanto de vídeo como de VoIP é de 100 ms, sendo este o atraso considerado tolerável por essas aplicações. Em termos de taxa de perda de pacote, o valor tolerável de de 10% [124].

Os cenários considerados apresentam um número de usuários ativos que varia de 10 a 100 usuários [119]. Variando-se o número de usuários, espera-se verificar como os algoritmos de escalonamento se comportam em termos de perda de pacotes, atraso de escalonamento e justiça, de acordo com o número de usuários presentes na célula.

Em todas as situações, todos os usuários aguardam a recepção dos três fluxos de informação, isto é, para um mesmo usuário é transmitido vídeo em tempo real, tráfego VoIP e uma sessão de navegação *web*. Dessa forma, espera-se conduzir os algoritmos a situações de sobrecarga, já que existem fluxos concorrentes a serem escalonados, e que aumentam na mesma proporção do número de usuários.

Tabela 5.1: Parâmetros do Canal SCM.

Parâmetro	**Valor**
Frequência de transmissão	2.0 GHz
Velocidade do terminal móvel	5.0 m/s e 40.0 m/s
Número de antenas na estação base	1
Número de antenas na estação móvel	1
Cenário	macrocélula suburbana
Número de multipercursos	19
Modelo de propagação	Okumura Hata

5.5.2 Ajuste dos Principais Algoritmos

Para a comparação dos resultados de escalonamento e desempenho da técnica analisada, foram considerados quatro algoritmos: máxima taxa, *proportional fair*, M-LWDF (*modified largest weighted delay first*) e o algoritmo do *framework* proposto nessa tese, identificado por CH-RL (*convex hull reinforcement learning*), em referência direta à tecnica de aprendizado de máquina utilizada para a resolução do problema de escalonamento e seleção de usuários. Nesta seção, são apresentados os ajustes de parâmetros de cada algoritmo, necessários para seu seu funcionamento apropriado.

Tabela 5.2: Parâmetros do sistema de transmissão

Parâmetro	**Valor**
Largura de banda	10 MHz
Espaçamento entre as subportadoras	15 KHz
Potência de trasmissão (estação rádio base)	43 dBm
Raio das células	500 m
Tamanho do *cluster*	4 células (reúso)
Total de células	21
Tamanho do *buffer* de transmissão	3000 bits
Trafego 1 (maior prioridade)	*stream* de vídeo (com codificação H.264)
Trafego 2 (prioridade média)	VoIP
Trafego 3 (menor prioridade)	Navegação *Web*

Tabela 5.3: Esquemas de Modulação e Codificação

MCS	**Modulação**	**Taxa de codificação**
1	QPSK	1/3
2	QPSK	1/2
3	QPSK	2/3
4	16QAM	1/2
5	16QAM	2/3
6	16QAM	4/5
7	64QAM	2/3
8	64QAM	4/5

5.5.3 Máxima Taxa

No algoritmo de máxima taxa, é calculada, dentro de um TTI, para cada usuário, a seguinte métrica [119]:

$$m_{i,k}^{MT} = \log\left(1 + SINR_k^i(t)\right) \tag{5.16}$$

em que $SINR_k^i(t)$ representa a razão sinal ruído mais interferência do i-ésimo usuário caso este seja alocado no k-ésimo *resource block* do TTI do instante t. Logo, $m_{i,k}^{MT}$ representa o maior valor possível de eficiência espectral esperado, de acordo com a fórmula de capacidade de Shannon. É selecionado para transmissão no instante t e *resource block* k o usuário i que apresenta o maior valor para a métrica $m_{i,k}^{MT}$. Observa-se que o algoritmo não leva em consideração qualquer informação sobre a aplicação do usuário. Seu objetivo é a seleção dos usuários que apresentam, potencialmente, as maiores taxas de transmissão.

5.5.4 Proportional Fair

No algoritmo *proportional fair*, é calculada, dentro de um TTI e para cada usuário, a métrica [126]:

$$m_{i,k}^{PF} = \frac{1}{\overline{R^i}(t-1)} \times \log\left(1 + SINR_k^i(t)\right) \tag{5.17}$$

em que $\overline{R^i}(t-1)$ representa o valor de vazão média do usuário até o instante $t-1$. A idéia do algoritmo é utilizar a vazão média do usuário como um fator que pondera a taxa de transmissão esperada, de forma que mesmo os usuários que apresentam condições de propagação desfavoráveis serão enventualmente escalonados. Dessa forma, percebe-se que o algoritmo procura ser justo na distribuição dos recursos de transmissão, sem privilegiar usuários com melhores condições de canal, diferentemente do que ocorre com o algoritmo de máxima taxa. É selecionado para transmissão no instante t e *resource block* k o usuário i que apresenta o maior valor para a métrica $m_{i,k}^{PF}$.

5.5.5 M-LWDF

No algoritmo M-LWDF, computa-se, dentro de um TTI, para cada usuário, a seguinte métrica [127]:

$$m_{i,k}^{M-LWDF} = -\frac{\log \delta_i}{\tau_i} \times D_{HOL,i} \times \frac{1}{\overline{R^i}(t-1)} \times \log\left(1 + SINR_k^i(t)\right) \tag{5.18}$$

em que δ_i representa a taxa de perda de pacote tolerável para o usuário i, τ_i representa o valor limite de atraso suportado pela aplicação do usuário i e $D_{HOL,i}$ representa o atraso do pacote que se encontra no início do *buffer* de transmissão do usuário i.

Percebe-se que a métrica utilizada pelo M-LWDF é, em parte, uma combinação das métricas dos algoritmos máxima taxa e *proportional fair*, de forma que busca-se maximizar a taxa de transmissão e garantir certa justiça na distribuição dos recursos. Cabe observar ainda que a introdução dos fatores δ_i e τ_i faz com que o algoritmo M-LWDF seja dependente da aplicação (*application aware*), ou seja, seu objetivo não é apenas procurar maximizar a taxa de transmissão ou buscar equidade na distribuição dos recursos de rádio, mas procura atender a métricas da própria aplicação, como perda de pacote e atraso máximo tolerável.

Para as simulações realizadas, tomou-se $\tau_i = 100$ ms e $\delta_i = 10^{-2}$ tanto para o tráfego de voz (VoIP) quanto para o tráfego de vídeo, conforme sugerido por [124]. Para o tráfego gerado pela navegação *web*, fez-se $\tau_i = 300$ ms e $\delta_i = 10^{-1}$ [124].

5.5.6 Ajuste do Framework Proposto (CH-RL)

Esta seção propõe-se a detalhar escolha dos parâmatros referentes às recompensas do algoritmo de seleção e escalonamento de usuário proposto nessa tese.

Conforme já abordado, é o sinal de recompensa, em um problema de aprendizado por reforço, que informa ao agente inteligente o quão apropriada foi sua ação em função do estado em que se encontra o sistema, e como essa ação interfere na evolução temporal desse mesmo sistema. Como trata-se de um problema multi-objetivo, o sinal de recompensa é multidimensional. Mais especificamente, o algoritmo proposto, CH-RL, apresenta três dimensões de recompensa.

A primeira dimensão refere-se à eficiência espectral da ação tomada, e é dada por

$$r_1(s_t, a_{ij}) = \frac{\rho\left(a_{ij}\right)}{\max\limits_{k} \rho\left(MCS_k\right)} \tag{5.19}$$

em que, como já anteriormente apresentado, $\rho\left(a_{ij}\right)$ representa a eficiência espectral máxima que pode ser atingida ao se transmitir utilizando o i-ésimo *resource block* do *slot* localizado no instante de tempo $j\Delta t$, supondo a alocação de potência uniforme entre todos os *resource blocks*, e $\rho\left(MCS_k\right)$ representa a máxima eficiência espectral disponível ao sistema (isto é, representa a eficiência espectral do esquema de modulação e codificação que possui modulação de ordem mais alta e maior taxa de codificação).

Diferentemente dos algoritmos considerados, a eficiência espectral é calculada tendo como base os valores fornecidos pelos esquemas de AMC disponíveis, e não pelo limite teórico de Shannon. Estes valores, por sua vez, são obtidos diretamente de outra aplicação de aprendizado por reforço, devidamente apresentada no capítulo 3, sobre modulação e codificação adaptativas. Utilizando essa abordagem, espera-se reduzir o hiato observado na capacidade de transmissão ao se utilizar as tabelas de consulta, tornando o valor de recompensa o mais próximo possível da vazão real, e não apenas uma aproximação utilizada pelas tabelas de consulta.

A segunda dimensão de recompensa refere-se ao atraso de transmissão, e é dada por

$$r_2(s_t, a_{ij}) = \begin{cases} 1, & \text{se } HOL_{t+j\Delta t} \leq T_{limite} \\ e^{-\beta_1\left(HOL_{t+j\Delta t} - T_{limite}\right)}, & \text{caso contrário} \end{cases} \tag{5.20}$$

em que escolhe-se $T_{limite} = 50$ ms, e $\beta_1 = 100$. Dessa forma, o algoritmo possui uma tolerência de 50 ms para escalonar um pacote (metade do valor limite de atraso tolerável pelas aplicações de VoIP e vídeo) e, no limite do atraso tolerável pelas aplicações sensíveis ao atraso, a recompensa recebida é praticamente nula. Os valores foram obtidos após alguns testes realizados, e essa combinação gerou resultado de escalonamento para os quais o agente foi capaz de escalonar os pacotes antes do atraso máximo suportado pelas aplicações, mas ainda assim considerando uma certa tolerência, de forma a não tomar decisões de escalonamento precipitadas, muito urgentes

ou que possam ocasionar elevado atraso de entrega. O comportamento da recompensa, a título de ilustração, é mostrado na Fig. 5.6.

Para aplicações que não são sensíveis ao atraso, como é o caso da navegação *web*, escolheu-se $r_2(s_t, a_{ij}) = 0,1$, ou seja, uma prioridade menor do que as aplicações sensíveis ao atraso e que se encontram com atraso de entrega de até 75 ms (75% do máximo tolerável), conforme pode ser observado ao se comparar o valor proposto com a recompensa mostrada na Fig. 5.6.

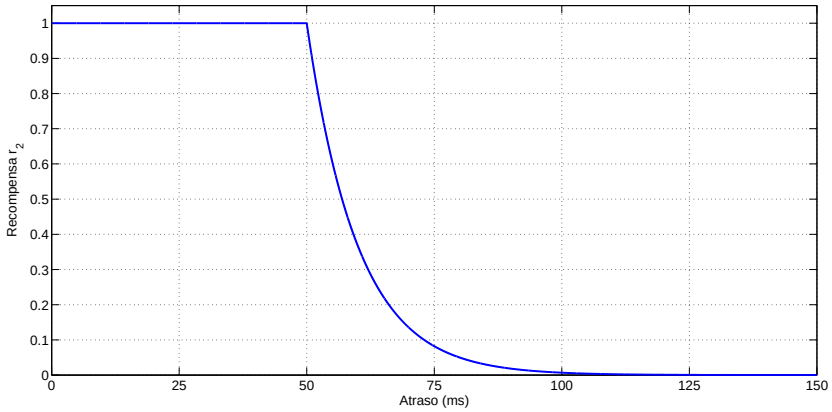


Figura 5.6: Ilustração do comportamento da recompensa na dimensão do atraso de escalonamento para o algoritmo CH-RL.

Em sua terceira dimensão, que diz respeito à ocupação do *buffer*, a recompensa é:

$$r_3(s_t, a_{ij}) = \begin{cases} 1, & \text{se } B_{t+j\Delta t} \leq B_{limite} \\ e^{-\beta_2 \left(B_{t+j\Delta t} - B_{limite}\right)}, & \text{caso contrário} \end{cases} \tag{5.21}$$

Para os valores do cenário de simulação, $B_{limite} = 1500$ bits (metade do tamanho do *buffer* disponível) e $\beta_2 = \frac{1}{300}$, de tal forma que o agente só começa a ser penalizado quando o *buffer* possui ocupação superior a 50%, e a recompensa é praticamente nula caso este se encontre completamente ocupado.

Para a combinação dos valores de recompensa, $R(s_t, a_{ij}) = \alpha_1 r_1(s_t, a_{ij}) + \alpha_2 r_2(s_t, a_{ij}) + \alpha_3 r_3(s_t, a_{ij})$, escolheram-se os valores das constantes como $\alpha_1 = 0,2$; $\alpha_2 = 0,5$ e $\alpha_3 = 0,3$, como forma de priorizar as aplicações sensíveis ao atraso. Esses valores foram obtidos após testes exaustivos entre diferentes combinações. A tripla apresentada foi a que apresentou como resultado a maior equidade na alocação de recursos e menor perda de pacotes de vídeo e de VoIP (parâmetros cujo significado ainda serão apresentados nas próximas seções). Entretanto, cabe observar que outras técnicas e estratégias de ajuste poderiam ser utilizadas (algumas dessas inclusive poderiam se valer de abordagens de aprendizado de máquina). Para os propósitos e escopo desse trabalho, a abordagem já foi suficiente para permitir a comparação, sob diferentes situações, dos algoritmos de escalonamento.

Para a predição dos valores de canal e de taxa de ocupação de *buffer*, foram utilizados filtros adaptativos com base no algoritmo de adaptação *set-membership affine projection*, utilizando filtros com 15 coeficientes e valor máximo de erro $\nu = 0,1$, conforme a Eq. (5.12).

5.5.7 Resultados

5.5.7.1 Convergência e Complexidade Computacional

A Fig. 5.7 mostra o tempo de convergência da abordagem proposta. A convergência foi medida observando-se as modificações na tabela de valores Q durante a fase de aprendizado do algoritmo. O valor mais próximo de zero indica que as iterações não provocavam mais mudanças nos valores da função estado-ação. Como é mostrado, o aprendizado da melhor política leva aproximadamente 100000 iterações, o que corresponde a um intervalo de tempo de 100 segundos, ou cerca de dois minutos, utilizando a estrutura de quadros de transmissão do padrão LTE, já descrita anteriormente.

Considerando a necessidade de operação em tempo real, algumas observações devem ser feitas. Em primeiro lugar, a solução que faz uso do aprendizado por reforço possui memória dos estados e, portanto, não precisa ser recalculada a cada novo início de transmissão, especialmente em ambientes de baixa variabilidade temporal. Parte das iterações pode ser realizada *off-line*, aproveitando os valores Q aprendidos durante a etapa inicial de convergência do algoritmo. Além disso, é possível que o terminal móvel troque informações de controle com a estação rádio-base mesmo quando não há transmissão de dados, de forma que o algoritmo CH-RL pode ser iniciado com informações *a priori*, reduzindo o tempo de convergência da proposta.

Quanto à convergência, ainda que o algoritmo de alocação proposto possua várias etapas (como predição de canal e busca pelos usuários na matriz de oportunidades de transmissão), o passo mais oneroso computacionalmente consiste na determinação do fecho convexo. Logo, a complexidade computacional do CH-RL pode ser aproximada por $|\mathcal{A}| \log |\mathcal{A}|$ [54], em que $\mathcal{A}$ é a cardinalidade do conjunto $\mathcal{A}$ de ações (para cada estado). Para a modelagem proposta, de acordo com a Eq. (5.2), é de $|\mathcal{A}| = 10N_{RB}$. Naturalmente, a proposta CH-RL é computacionalmente mais complexa que as abordagens clássicas e também exige mais memória, uma vez que deve ser armazenada a informação de um fecho convexo por estado por usuário, além do próprio vetor de estados.

5.5.7.2 Vazão Média da Célula

A Fig. 5.8 mostra a comparação dos algoritmos de escalonamento quanto à vazão total da célula. É possível observar que o escalonador por máxima taxa apresenta os resultados de vazão mais altos, uma vez que realiza a seleção e escalonamento dos usuários que apresentam a melhor resposta de canal para transmissão, em acordo com a Eq. (5.16). As demais estratégias conseguem

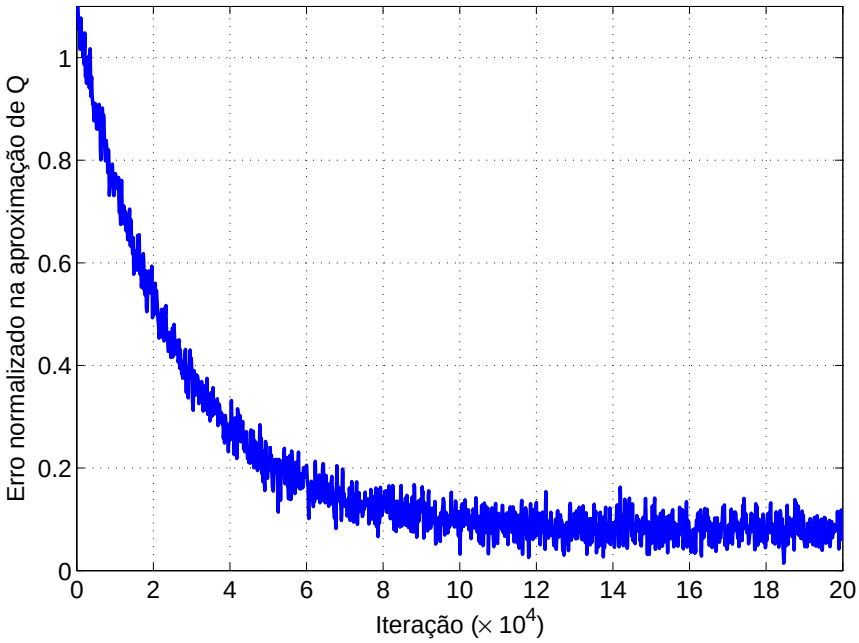


Figura 5.7: Curva de aprendizado da estratégia de escalonamento e seleção de usuários CH-RL.

garantir entre 40% e 50% do valor de vazão fornecido pelo algoritmo de máxima taxa.

Observa-se ainda que o algoritmo proposto nesta tese, CH-RL, apresenta desempenho ligeiramente superior às estratégias PF e M-LWDF em termos de vazão média da célula, pois a estratégia á capaz de derivar os valores de eficiência espectral a partir dos próprios esquemas de AMC, sem utilizar o limite de Shannon para a capacidade de transmissão.

É ainda possível perceber o efeito do ganho por diversidade multi-usuário. Nota-se a tendência de a capacidade da célula aumentar com o aumento do número de usuários na célula, e a explicação do fenômeno é simples: com o aumento do número de usuários, aumenta-se a probabilidade de, em um dado instante e em um dado conjunto de frequências, encontrar usuários que experimentam melhores condições de canal e de propagação. Naturalmente, o efeito da diversidade é mais pronunciado no algoritmo de escalonamento por máxima taxa, já que os outros algoritmos levam em consideração outros fatores que não apenas a maximização da taxa de transmissão.

5.5.7.3 Vazão Média por Usuário

A Fig. 5.9 ilustra a vazão média por usuário para os algoritmos de escalonamento considerados. Uma vez que uma quantidade fixa de recursos de rádio deve ser alocada para os usuários, aumentar o número de usuários implica necessariamente em diminuir a vazão média de cada usuário. O escalonador por máxima taxa apresenta, em termos de taxa de transmissão, o melhor desempenho quando comparado aos demais algoritmos. Entretanto, ele só é capaz de

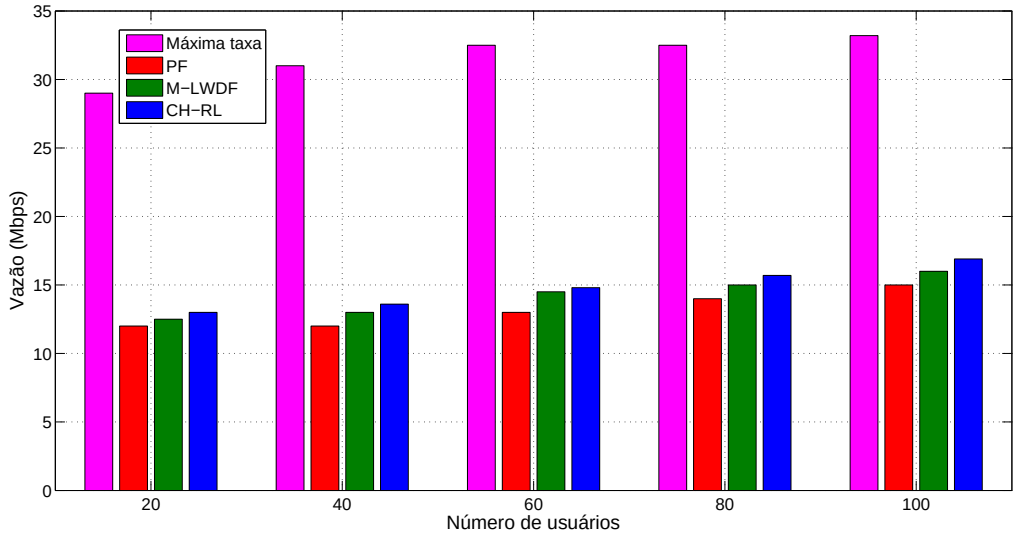


Figura 5.8: Vazão média da célula em função do número de usuários para as estratégias de escalonamento por máxima taxa, *proportional fair*, M-LWDF e CH-RL.

garantir maior taxa média para os usuários que aresentam as melhores condições de propagação, desfavorecendo usuários que se encontram nas bordas da célula e sendo incapaz de levar em consideração qualquer informação sobre o tipo de aplicação de cada usuário, uma vez que esta não foi definida em termos de outras métricas de qualidade de serviço.

Repete-se ainda a tendência do algoritmo proposto apresentar desempenho ligeiramente superior às abordagens de escalonamento definidas pelos algoritmos *proportional fair* e M-LWDF.

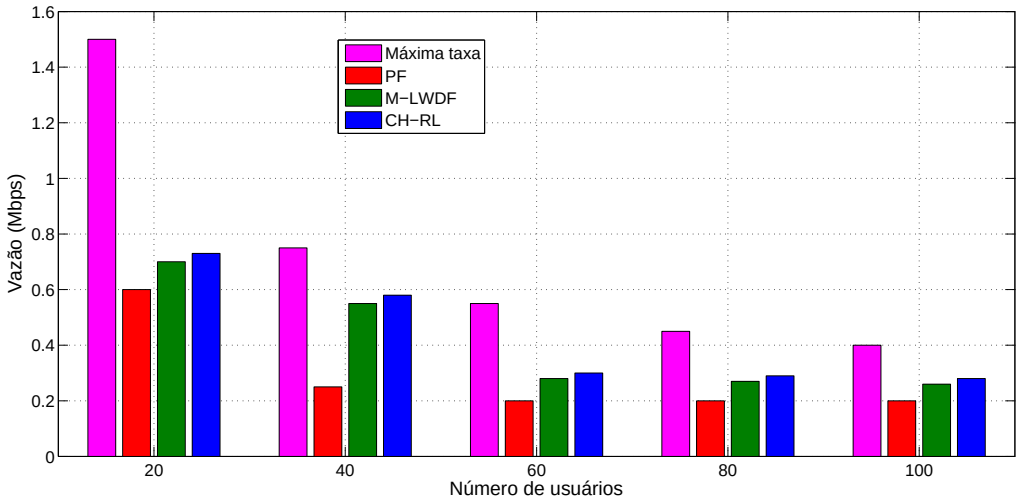


Figura 5.9: Vazão média por usuário em função do número de usuários para as estratégias de escalonamento por máxima taxa, *proportional fair*, M-LWDF e CH-RL.

5.5.7.4 Justiça (Fairness)

Como é sabido, maximizar a capacidade de transmissão não deve ser o único objetivo em um sistema de comunicação multi-usuário. Conforme exposto anteriormente, um algoritmo de escalonamento deve buscar atender a outras métricas, como os requisitos de QoS dos usuários e a justiça (em termos de equidade) na distribuição dos recursos de transmissão.

Para determinar a justiça dos algoritmos na alocação dos recursos de rádio, foi utilizado o índice de Jain [128], definido por:

$$J(x) = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \sum_{i=1}^{N} (x_i)^2} \tag{5.22}$$

em que x representa o recurso cuja distribuição é avaliada, e N é o tamanho da população de indivíduos que solicita por esse recurso. O valor do índice de Jain varia entre $\frac{1}{N}$ (distribuição mais injusta) e 1 (distribuição igualitária).

A Fig. 5.10 resume o cálculo do índice de Jain para os algoritmos de escalonamento considerados, em que o recurso x em questão é a taxa de transmissão alcançada por um usuário, e N é o número de usuários na célula. É possível perceber que o algoritmo de máxima taxa possui o pior desempenho em termos de justiça entre os algoritmos analisados. A explicação do fato é simples: o algoritmo de escalonamento por máxima taxa é capaz de garantir vazão elevada apenas para um conjunto bastante limitado de usuários (aqueles que possuem melhores condições de canal), enquanto os demais usuários experimentam baixas taxas de transmissão. O cálculo do índice apenas evidencia o resultado já exposto nas seções anteriores.

O algoritmo *proportional fair* apresenta o índice de justiça mais alto, uma vez que considera, em sua formulação, a justiça na distribuição dos recursos. Até certo ponto, ele é invariante ao número de usuários, sendo capaz de garantir índices de justiça semelhantes mesmo com o aumento do número de usuários. Em contrapartida, sua vazão média é a menor entre os algoritmos considerados, conforme mostrado nas Figs. 5.8 e 5.9.

Os algoritmos M-LWDF e CH-RL apresentam melhor índice de justiça do que a estratégia por máxima taxa, mas desempenho inferior ao algoritmo *proportional fair*, uma vez que eles não visam apenas justiça na alocação dos recursos, mas levam em conta em sua formulação características da aplicação de cada usuário, como será evidenciado nas próximas seções.

Como é possível inferir a partir dos resultados apresentados até o momento, o algoritmo de máxima taxa, apesar de maximizar a capacidade do sistema, não é capaz de garantir uma distribuição justa dos recursos de rádio entre os usuários. Consequentemente, não será capaz de prover qualidade de serviço para tráfegos que dependem de métricas como taxa de erro de pacote e atraso sofrido pelo pacote,uma vez que só leva em consideração a vazão de cada usuário. Por

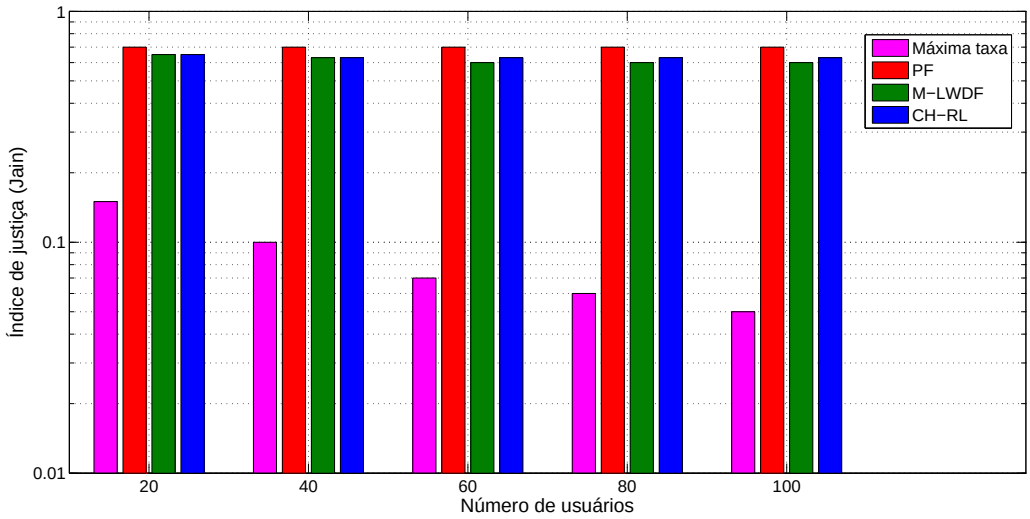


Figura 5.10: Índice de justiça (Jain) na alocação dos recursos de rádio em função do número de usuários para as estratégias de escalonamento por máxima taxa, *proportional fair*, M-LWDF e CH-RL.

esse motivo, nas próximas seções, o algoritmo de máxima taxa será omitido da análise. Serão considerados apenas as formas de escalonamento *proportional fair*, M-LWDF e CH-RL. Ainda que o *proportional fair* não seja voltado para atender métricas específicas de aplicações, seu comportamento servirá como referência para a comparação dos algoritmos M-LWDF e CH-RL.

5.5.7.5 Taxa de Perda de Pacotes

Uma das formas de se compararem os algoritmos de escalonamento é por meio da taxa de perda de pacotes. As Figs. 5.11 e 5.12 mostram a taxa de perda de pacotes de vídeo para cada um dos algoritmos considerados em dois cenários distintos: baixa mobilidade (estação móvel a 5.0 m/s) e alta mobilidade (estação móvel a 40.0 m/s).

Antes de prosseguir com a análise, cabe salientar dois fatores: o primeiro é que um pacote é considerado perdido caso tenha um atraso de entrega superior ao limite tolerável pela aplicação (100 ms no caso de vídeo e VoIP) ou experimente condições de propagação que gerem uma taxa de erro de bloco (BLER) superior a 10^{-1} [112]. A aplicação de navegação *web* foi desconsiderada da análise uma vez que não é sensível ao atraso. o que não ocorre com o vídeo em tempo real. O segundo fator diz respeito aos resultados obtidos para as aplicações sensíveis ao atraso. Estes foram semelhantes em termos de taxa de perda de pacote e atraso na entrega de pacote. Sendo assim, serão mostrados apenas os resultados referentes ao tráfego de vídeo, sendo válido comportamento semelhante para VoIP.

Um aumento no número de usuários provoca um aumento na taxa de perda de pacotes de vídeo, uma vez que a mesma quantidade de recursos de rádio deve ser alocada para um número

maior de estações que os solicitam. O algoritmo *proportional fair* preocupa-se exclusivamente em garantir a equidade na distribuição dos recursos, de forma que pacotes que exigem maior urgência no escalonamento (pois o atraso experimentado está próximo ao limite tolerável pela aplicação) não serão considerados pelo algoritmo, e acabarão por ser descartados.

Por outro lado, as estratégias de escalonamento M-LWDF e CH-RL levam em consideração o atraso já experimentado por um pacote enquanto este aguarda uma oprtunidade de transmissão, de forma que a taxa de perda de pacotes é consideravelmente menor do que a observada pelo algoritmo *proportional fair*.

Em termos do desempenho quanto à variação da velocidade, percebe-se que o algoritmo *proportional fair* é invariante a este efeito, uma vez que não leva em consideração a resposta do canal para o escalonamento dos usuários. Em cenários de baixa mobilidade, o M-LWDF e o CH-RL apresentam deempenho semelhante. Com o aumento da velocidade, o algoritmo proposto passa a se destacar, apresentando menor perda que o M-LWDF. Este comportamento é justificado, em primeiro lugar, pela utilização da predição da resposta de canal, de forma a trabalhar com informação atualizada sobre o estado de canal. Em segundo lugar, tem-se a utilização de um horizonte de escalonamento maior que um TTI, o que fornece ao algoritmo a liberdade para escolher o melhor instante de escalonamento dos usuários, tendo comportamento menos míope que os demais algoritmos e sendo capaz de identificar, por meio da predição, possíveis degradações na resposta do canal, escalonando com mais urgência os usuários para os quais essa situção é verificada.

Em resumo, ressalta-se a menor perda de pacotes do algoritmo proposto. Uma vez que, como o horizonte de decisão de escalonamento do CH-RL é superior a um TTI, o algoritmo é capaz de identificar as melhores oportunidades de transmissão (em termos de resposta de canal) sem atrasar de forma excessiva a entrega do pacote ou sobrecarregar o *buffer* de transmissão, o que não ocorre com o algoritmo M-LWDF, que escalonaria um pacote mesmo em condições de propagação desfavoráveis, o que causaria a eventual perda desse pacote.

5.5.7.6 Atraso na Entrega de Pacotes

As Figs. 5.13, 5.15, 5.17 e 5.19 mostram o desempenho dos algoritmos com relação ao tempo de entrega dos pacotes presentes no *buffer* de transmissão para um ambiente de baixa mobilidade, e as Figs. 5.14, 5.16, 5.18 e 5.20 mostram o desempenho dos algoritmos com relação ao tempo de entrega dos pacotes presentes no *buffer* de transmissão quando em um ambiente de alta velocidade. Mais especificamente, é mostrada, para todos os casos analisados, a densidade de probabibilidade acumulada (CDF) para o tempo de entrega de um pacote de vídeo.

Considerando o efeito da velocidade, infere-se que, para ambientes que apresentam usuários de baixa mobilidade, os algoritmos M-LWDF e CH-RL apresentam comportamento semelhante e, com o aumento da mobilidade, o CH-RL tende a apresentar melhor desempenho que o M-

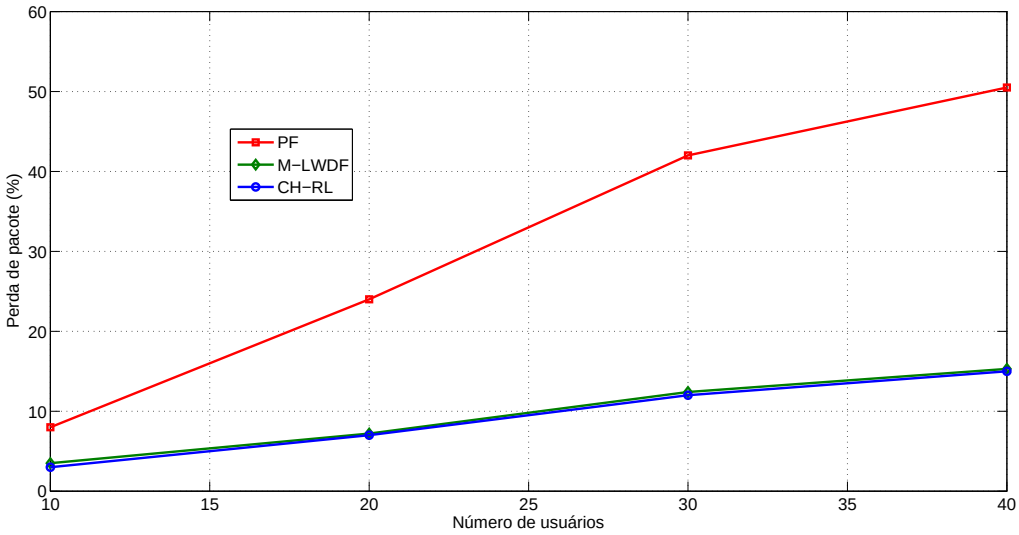


Figura 5.11: Taxa de perda de pacotes de vídeo em função do número de usuários para as estratégias de escalonamento *proportional fair*, M-LWDF e CH-RL em um cenário de baixa mobilidade.

LWDF. Uma vez mais, o *proportional fair* possui desempenho invariante à velocidade da estação móvel. A justificativa para esse comportamento já foi explorada na seção anterior, e diz respeito à predição de canal e ao horizonte para tomada de decisão de escalonamento.

Quanto ao número de usuários, percebe-se que quando este é reduzido, os algoritmos se comportam de maneira semelhante (Figs. 5.13 e 5.14), uma vez que não há contenda acirrada pelos recursos de rádio. À medida que o número de usuários aumenta, percebe-se que o algoritmo *proportional fair* é incapaz de garantir o tempo limite e entrega de pacote de 100 ms para todos os usuários. Esse já era um fato esperado, pois o algoritmo visa apenas equidade na distribuição da taxa de transmissão, não sendo apropriado para lidar com tipos de tráfego que possuem restrição quanto ao atraso de transmissão.

Já os algoritmos CH-RL e M-LWDF são cientes das necessidades específicas da aplicação. Ambos são capazes de entregar os pacotes dentro do limite de atraso máximo suportados pela aplicação de vídeo. Destaca-se a superioridade do algoritmo proposto, sobretudo em cenários de alta mobilidade, nos quais ele é capaz de escalonar o tráfego de vídeo com maior folga do que o M-LWDF, mesmo com o aumento do número de usuários. O CH-RL é capaz de analisar um horizonte maior de decisão de escalonamento, além de monitorar o *buffer* de transmissão, justificando seu desempenho superior com relação às outras duas estratégias consideradas. Como mostra, por exemplo, a Fig. 5.20, a diferença entre o percentual de pacotes escalonados em até 40 ms pode chegar a 20%, o que representa um ganho de aproximadamente 33% do algoritmo proposto com relação à estratégia M-LWDF.

É perceptível a degradação do algoritmo M-LWDF com o aumento do número de usuários e da

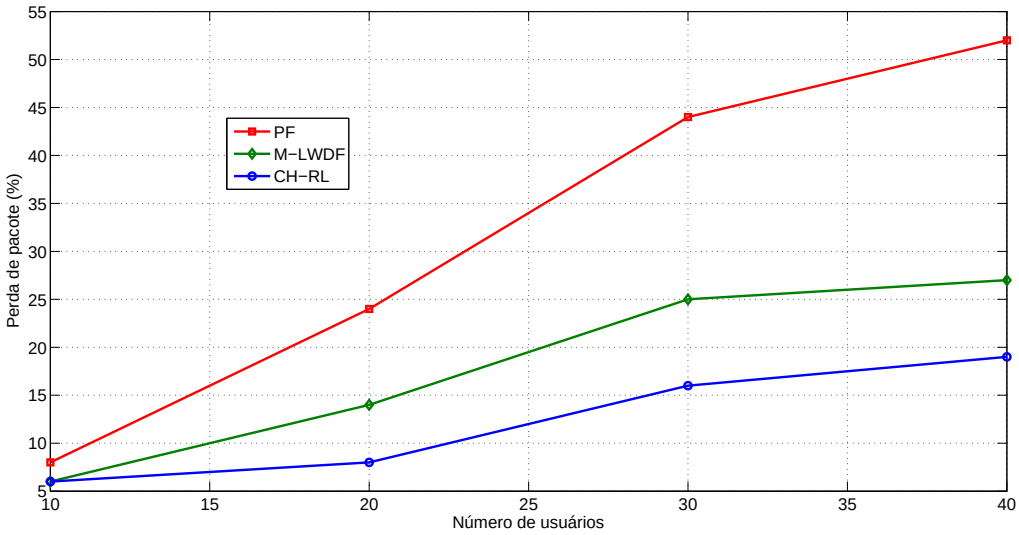


Figura 5.12: Taxa de perda de pacotes de vídeo em função do número de usuários para as estratégias de escalonamento *proportional fair*, M-LWDF e CH-RL em um cenário de alta mobilidade.

mobilidade dos terminais, uma vez que este não é capaz de acompanhar a variabilidade maior do canal de comunicação com o aumento da velocidade dos móveis, enquanto que a proposta CH-RL é capaz de tirar proveito desse fato por meio do horizonte maior de decisão de escalonamento e da informação atualizada sobre o estado de canal, que é fornecida pelo preditor de canal. Este fato é verificado observando-se que o desempenho do algoritmo permanece quase inalterado quando o número de usuários é mantido constante e a velocidade é variada. Por outro lado, ainda que haja a degradação do desempenho do algoritmo CH-RL à medida que o número de usuários aumenta, esta é menos acentuada do que a observada na estratégia M-LWDF.

5.5.7.7 Influência da Predição de Canal

A fim de promover uma comparação mais justa entre a abordagem proposta e o algoritmo M-LWDF, considera-se também o caso em que o algoritmo M-LWDF opera conjuntamente com o preditor de resposta do canal, de forma a possuir informação sempre atualizada sobre o estado do canal. As Figs. 5.21, 5.22 e 5.23 mostram a CDF do atraso de pacotes de vídeo em cenário de alta mobilidade para a modificação proposta.

Como é possível inferir, a informação atualizada sobre o estado do canal exerce impacto sobre o desempenho do algoritmo M-LWDF, reduzindo o ganho que a abordagem CH-RL possui sobre o M-LWDF. Entretanto, apenas a informação atualizada sobre o estado do canal não é suficiente para igualar o desempenho dos algoritmos. Nota-se que parte da superioridade da abordagem proposta provém do fato de observar o nível de ocupação do *buffer* de transmissão e de se tomar a decisão de escalonamento considerando o horizonte de tempo de todo o quadro de transmissão,

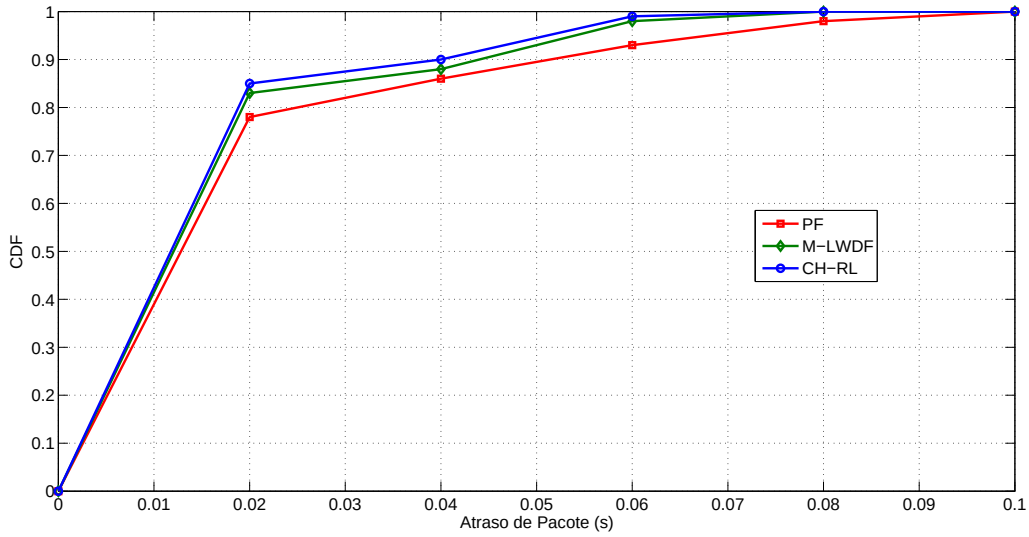


Figura 5.13: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de baixa mobilidade com 10 usuários.

e não apenas do próximo TTI, como é o caso do M-LWDF.

5.5.7.8 Classe de Serviço Best Effort

Como última análise, é interessante ainda notar que os algoritmos baseados em métricas de QoS acabam por direcionar a maior parte dos recursos alocados para o escalonamento do tráfego com restrições de atraso, em detrimento da aplicação de navegação *web*, que acabam por ser atendidas segundo uma política *best effort*.

A Fig. 5.24 mostra a vazão agregada do tráfego gerado pela navegação *web* em função do número de usuários. Constata-se que, em virtude do aumento do tráfego de vídeo, há diminuição do número de recursos destinados à navegação *web* e, consequentemente, diminuição da vazão agregada para esse tipo de serviço. Verifica-se a tendência dos algoritmos M-LWDF e CH-RL de priorizar os recursos para o tráfego do tempo real, por apresentar exigências maiores de qualidade de serviço.

Entretanto, observa-se que o algoritmo proposto é capaz de lidar melhor com o aumento do número de usuários, uma vez que uma das dimensões de análise é também a ocupação do buffer, de forma que o tráfego que provém da navegação *web* será ocasionalmente escalonado a partir do momento no qual passa a haver diminuição do valor de recompensa para a respectiva dimensão analisada.

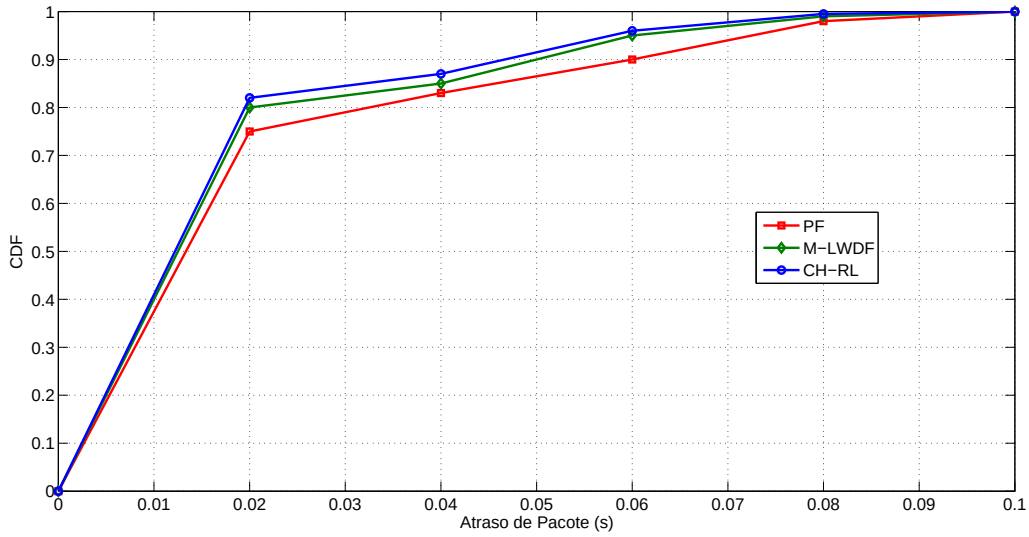


Figura 5.14: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de alta mobilidade com 10 usuários.

5.5.7.9 Usuários com Taxas de Transmissão Diferenciadas

Foi realizado ainda um último experimento, no qual o tráfego gerado pela navegação *web* foi substituído pela transmissão de três fluxos FTP [112, 124], com taxas médias teóricas de transmissão de 1 Mbps, 2 Mbps e 5 Mbps, em um cenário com 10 usuários. Com esse teste, pretendeu-se avaliar a capacidade do algoritmo M-LWDF e da estratégia CH-RL em atender usuários que possuem diferentes requisitos de taxa de transmissão em serviços que não apresentam característica de tempo real. Os três fluxos poderiam representar, em um cenário real, usuários que utilizam o serviço de *download* por meio de diferentes planos de serviço.

Os resultados, que representam a função de densidade de probabilidade acumuladas das taxas alcançadas, são mostrados nas Figs. 5.25, 5.26 e 5.27, para as taxas de transmissão de 1 Mbps, 2 Mbps e 5 Mbps, reespectivamente. Como evidenciado pelos gráficos, nenhuma das estratégias é capaz de garantir a vazão máxima de cada um dos fluxos para todos os usuários, uma vez que elas não foram concebidas com esse propósito. Esse fato é particularmente verdadeiro para a estratégia M-LWDF, cujo objetivo principal é garantir o escalonamento de tráfego sensível ao atraso, em tempo real.

É possível ainda observar que a estratégia proposta nessa tese, ainda que não seja capaz de garantir a vazão máxima para todos os fluxos FTP, apresenta resultados médios superiores ao algoritmo M-LWDF. Para o fluxo de 1 Mbps, o algoritmo CH-RL consegue garantir taxas entre 600 kbps e 750 kbps para 50% dos usuários, enquanto que o o algoritmo M-LWDF garante entre 400 kbps e 650 kbps para 50% dos usuários. Com o aumento da taxa média para 2 Mbps, o algoritmo CH-RL consegue garantir taxas entre 1,0 Mbps e 1,3 Mbps para 50% dos usuários, enquanto que o o algoritmo M-LWDF garante entre 200 kbps e 900 kbps para 50% dos usuários.

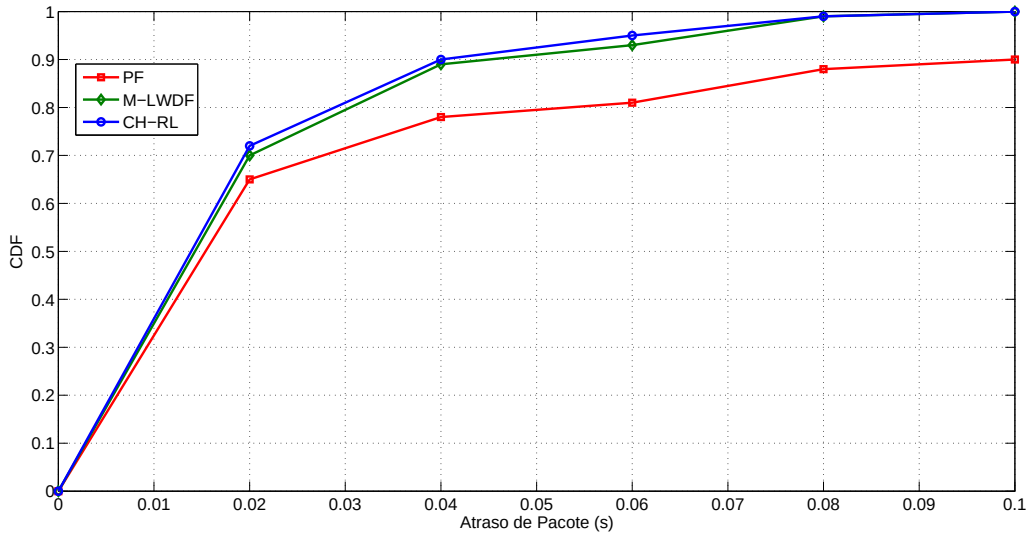


Figura 5.15: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de baixa mobilidade com 20 usuários.

Situação semelhante é observada para o fluxo de 5 Mbps: o algoritmo CH-RL consegue garantir taxas entre 2,0 Mbps e 2,5 Mbps para 50% dos usuários, enquanto que o o algoritmo M-LWDF garante entre até 1,75 Mbps para 50% dos usuários. Observa-se gradativamente a tendência de a taxa média ser reduzida, uma vez que transmissões a taxas mais altas exige um maior número de recursos de rádio.

Parte da superioridade do algoritmo proposto se deve a sua capacidade de observar o *buffer* de transmissão: quanto maior a taxa de geração de pacotes, mais rapidamente se dá o preenchimento do *buffer*, e essa urgência no escalonamento é identificada pela estratégia CH-RL. Outro efeito dessa política, além da maior taxa média de transmissão, é a menor variabilidade em termos de taxas alcançadas quando comparado ao algoritmo M-LWDF.

5.6 CONCLUSÕES

Este capítulo apresentou uma proposta de *framework* denominado CH-RL para o tratamento do problema de escalonamento de usuários em sistemas OFDMA, tomando como referência, em particular, uma estrutura de transmissão de quadros similar à utilizada no padrão LTE.

Inicialmente foram feitas considerações sobre os algoritmos de escalonamento e seleção de usuários atualmente utilizados em sistemas de comunicação multiusuário, levantando os principais requisitos que devem ser atendidos. Estes se resumem essencialmente em métricas de qualidade de serviço relacionadas com a taxa de transmissão, atraso de transmissão e taxa de perda de pacote.

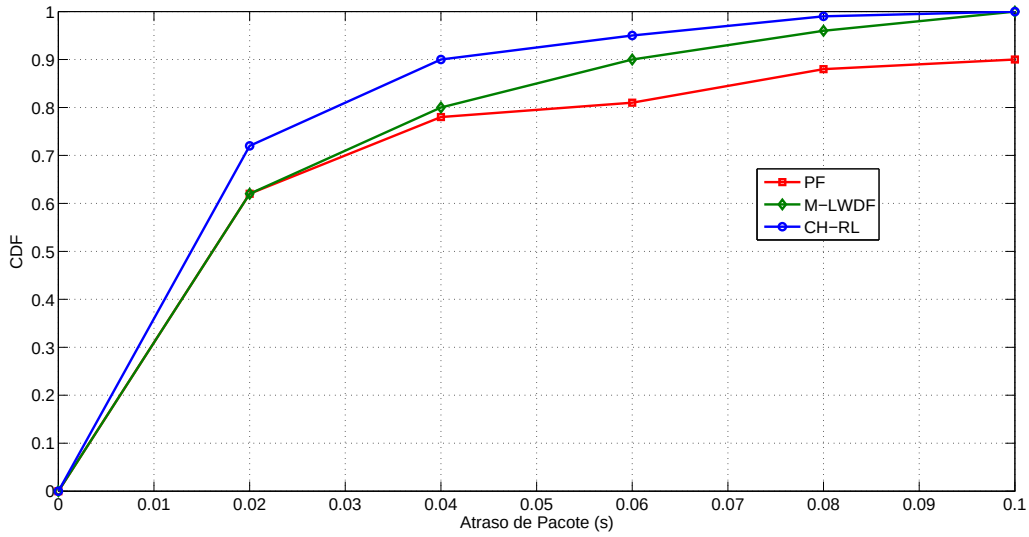


Figura 5.16: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de alta mobilidade com 20 usuários.

Em seguida, apresentou-se um *framework* para tratar o problema de alocação de recursos em sistemas mutliusuário utilizando a teoria de aprendizado por reforço multiobjetivo. Foram apresentadas as definições de estados, ações e recompensas do sistema de forma a traduzir os diferentes requisitos de qualidade de serviço, estes relacionados com a heterogeneidade das aplicações que trafegam nas redes de comunicação.

Em seguida, foram apresentados resultados de simulação que visavam comparar o desempenho do *framework* de escalonamento e alocação de recursos com algoritmos clássicos de escalonamento, a citar o algoritmo de máxima vazão, *proportinal fair* e M-LWDF. As simulações foram realizadas em nível sistêmico, utilizando as especificações do enlace direto do padrão 3GPP-LTE.

Os algoritmos foram compararados em termos de vazão, justiça, perda de pacotes e atraso na entrega de pacotes. Verificou-se que, apesar de o algoritmo de máxima taxa ser capaz de atingir os maiores valores para vazão da célula e taxa média de transmissão por usuário, não é capaz de garantir outras métricas de QoS, como atraso máximo tolerável, para aplicações sensíveis ao atraso, como é o caso do tráfego de *stream* de video em tempo real e VoIP.

Por outro lado, tanto o M-LWDF, que é orientado às necessidades de aplicações pouco tolerantes ao atraso, como o CH-RL, são capazes de atender esta métrica, sendo o algoritmo proposto superior à abordagem utilizada pelo M-LWDF. Isto se deve pelo fato de o algoritmo trabalhar com um horizonte maior para análise das decisões de escalonemento, possuindo consequentemente maior margem para lidar com a seleção e o escalonamento de usuários.

Os algoritmos foram ainda comparados em um cenário de taxas de transmissão diferenciadas de serviços não tempo real. Devido à sua capacidade de observação do enchimento do buffer de transmissão, a estratégia de escalonamento proposta, CH-RL, foi capaz de garantir taxas médias

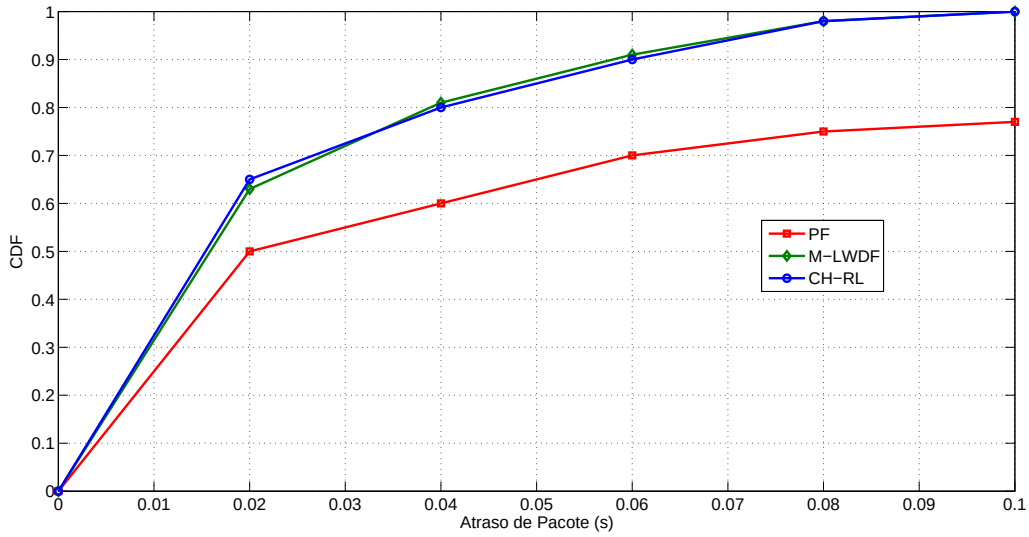


Figura 5.17: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de baixa mobilidade com 30 usuários.

de transmissão superiores àquelas observadas quando da utilização do algoritmo M-LWDF.

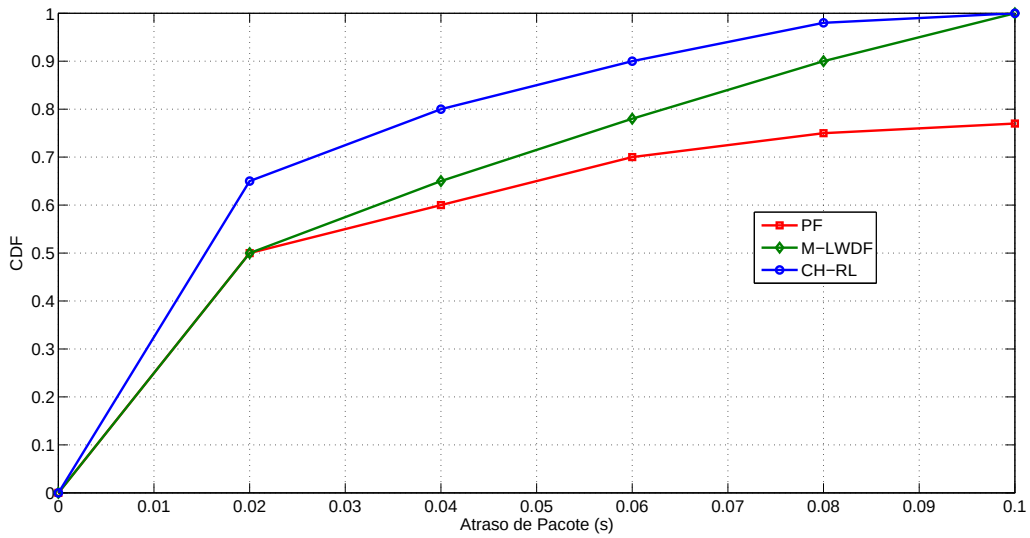


Figura 5.18: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de alta mobilidade com 30 usuários.

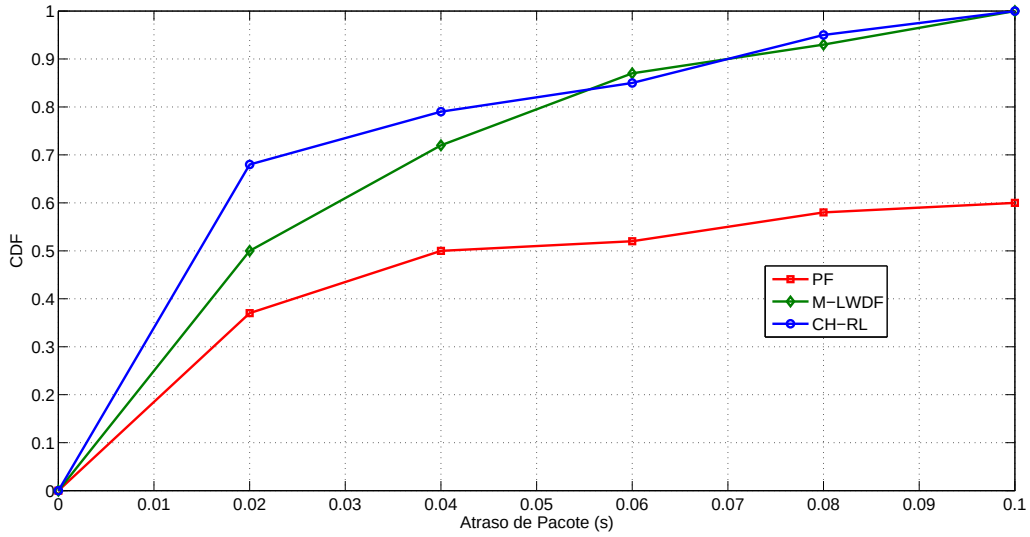


Figura 5.19: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de baixa mobilidade com 40 usuários.

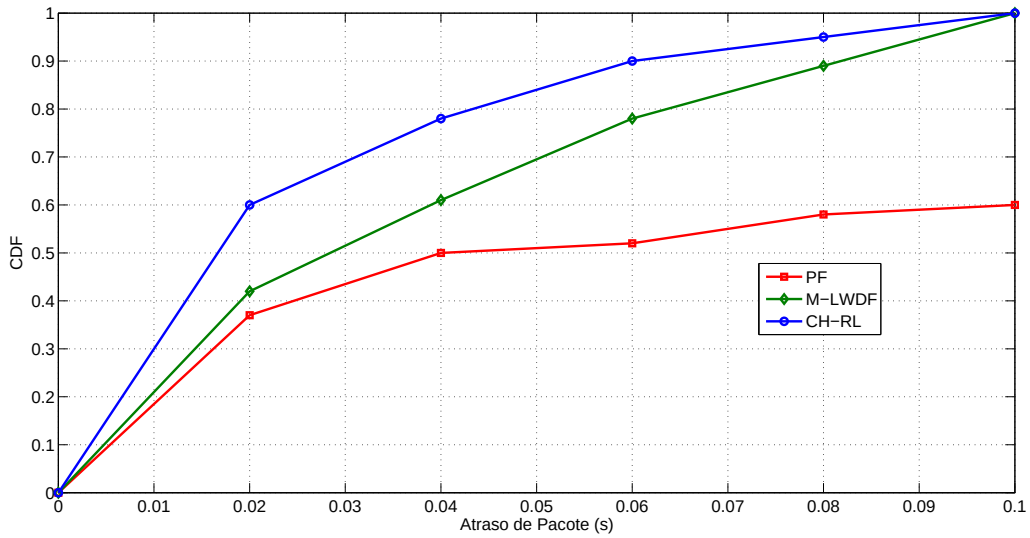


Figura 5.20: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento *proportional fair*, M-LWDF e CH-RL, para um cenário de alta mobilidade com 40 usuários.

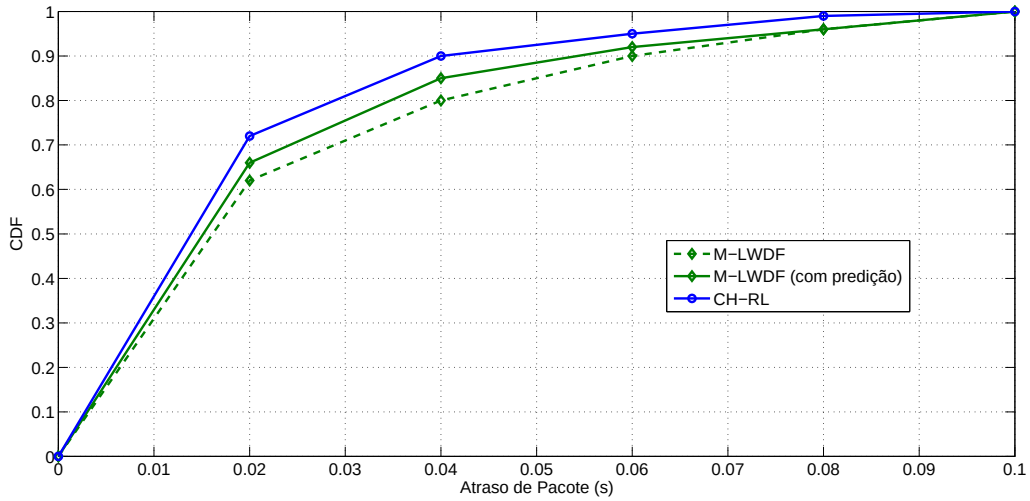


Figura 5.21: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento M-LWDF (com e sem predição de canal) e CH-RL, para um cenário de alta mobilidade com 20 usuários.

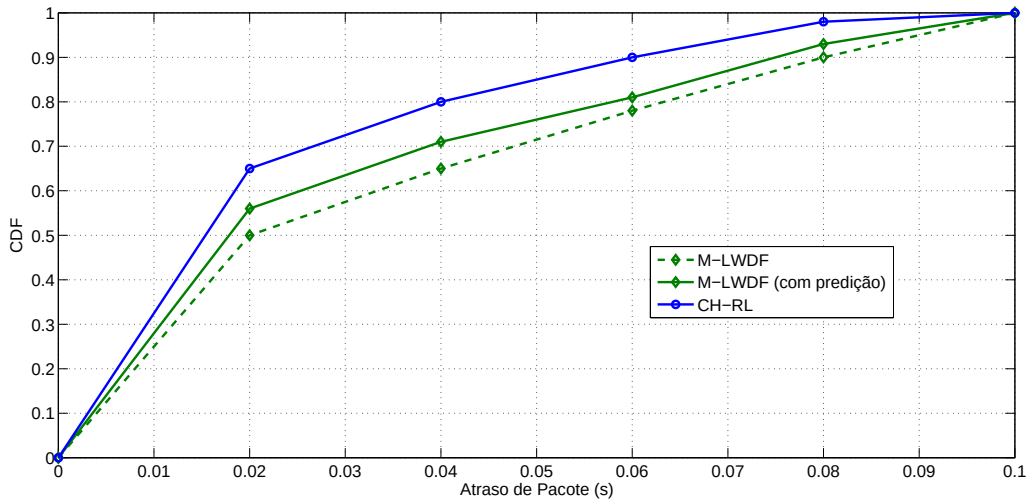


Figura 5.22: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento M-LWDF (com e sem predição de canal) e CH-RL, para um cenário de alta mobilidade com 30 usuários.

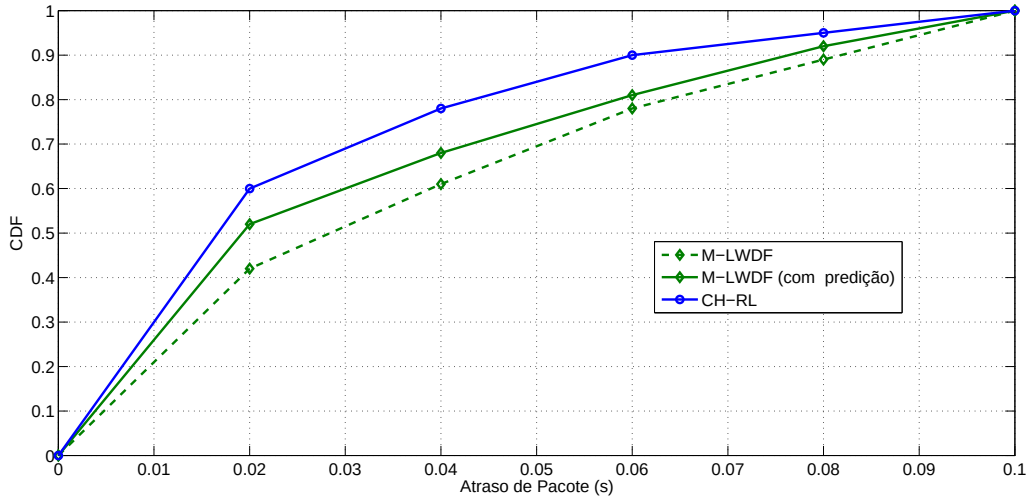


Figura 5.23: CDF do atraso de pacotes de vídeo, considerando os algoritmos de escalonamento M-LWDF (com e sem predição de canal) e CH-RL, para um cenário de alta mobilidade com 40 usuários.

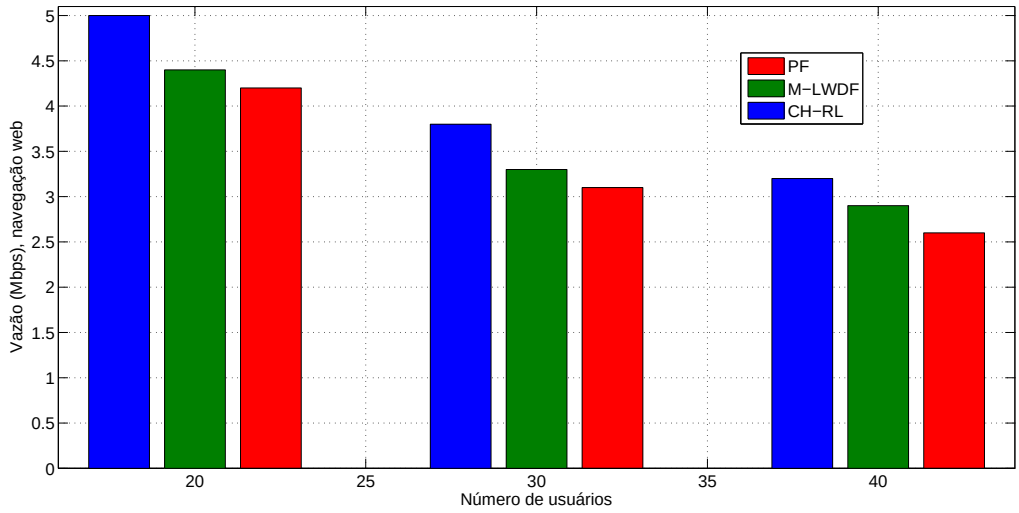


Figura 5.24: Vazão agregada para o tráfego do tipo navegação *web* em função do número de usuários, para os diferentes algoritmos de escalonamento considerados.

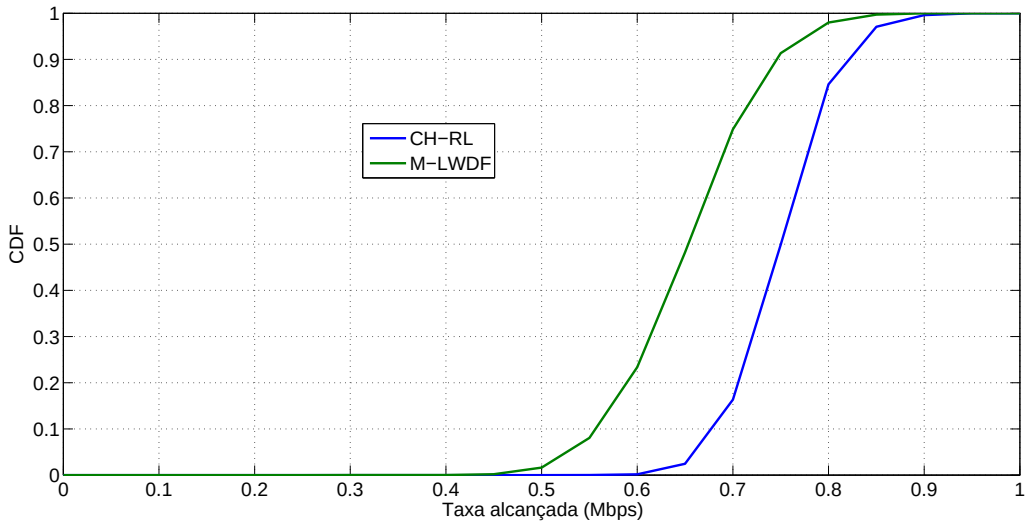


Figura 5.25: CDF da taxa de transmissão alcançada para um fluxo FTP de 1 Mbps, para um cenário de alta mobilidade com 10 usuários, comparando os algoritmos M-LWDF e CH-RL.

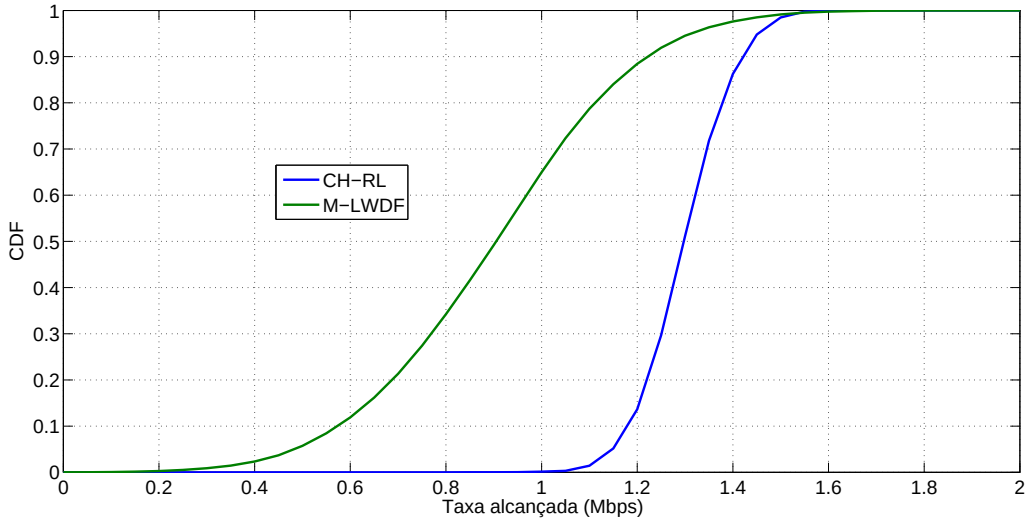


Figura 5.26: CDF da taxa de transmissão alcançada para um fluxo FTP de 2 Mbps, para um cenário de alta mobilidade com 10 usuários, comparando os algoritmos M-LWDF e CH-RL.

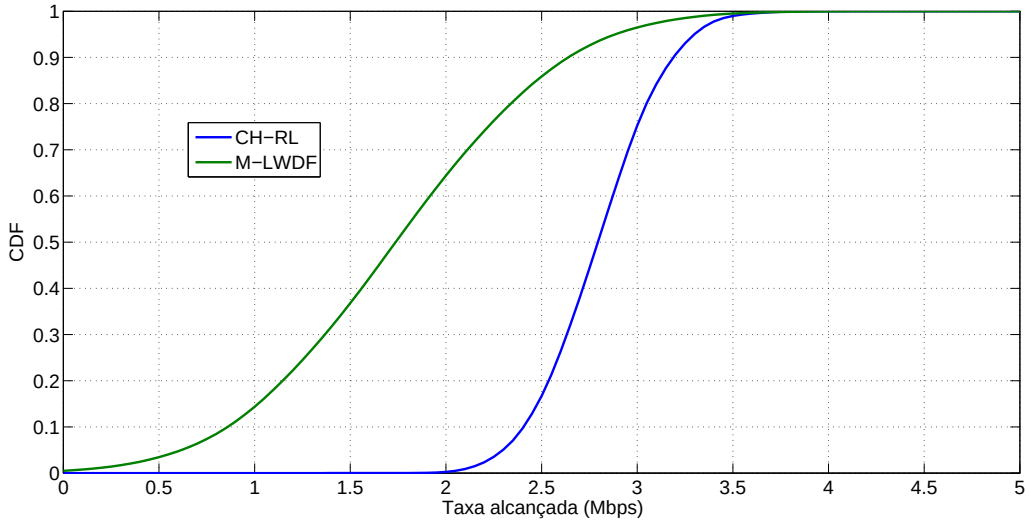


Figura 5.27: CDF da taxa de transmissão alcançada para um fluxo FTP de 5 Mbps, para um cenário de alta mobilidade com 10 usuários, comparando os algoritmos M-LWDF e CH-RL.

6 CONCLUSÕES

Este trabalho foi consgrado a a investigar a utilização de técnicas de aprendizado de máquina baseadas em aprendizado por reforço para a solução do problema de alocação de recursos e escalonamento de usuários no enlace direto de sistemas de comunicação digital baseados em OFDMA.

Inicialmente, ofereceu-se uma breve revisão bibliográfica e o formalismo necessário para a compreensão das técnicas de aprendizado por reforço sobre as quais esta tese versou. Foi tratado o problema de decisão Markoviano e como as informações que estão disponíveis ao projetista de um sistema de aprendizado por reforço influenciam na escolha dos algoritmos que podem ser utilizados para a solução do problema. Apresentaram-se as abordagens clássicas oferecidas pelos algoritmos de Diferenças Temporais, SARSA e *Q-learning*, além de uma vertente de problemas de aprendizado por reforço de estados contínuos, e sua solução por meio do algoritmo LSPI, e outra vertente de problemas do tipo multi-objetivo, que se utiliza do conceito de fecho convexo para a iteração dos valores Q em caso de recompensas multidimensionais.

Em seguida, foi tratado o problema de adaptação de enlace por meio da modulação e codificação adaptativas. Destacou-se o fato de que a abordagem clássica para a solução do problema de AMC, que utiliza tabelas de consulta, pode gerar soluções subótimas em termos de capacidade de transmissão, uma vez que são utilizados modelos teóricos idealizados de interfaces de transmissão, recepção e canal de comunicação. O capítulo apresentou como contribuições a modelagem da estratégia de AMC como um problema de aprendizado por reforço de estados contínuos, e a modificação do algoritmo LSPI para que este passe a operar em tempo real, atendendo às necessidades mais realistas do problema de adaptação de enlace. Resultados de simulação mostraram que a estratégia proposta oferece ganho com relação à abordagem clássica, sendo este ganho condicionado pela capacidade de aprendizado e adaptação que é fornecida por meio do aprendizado de máquina.

O próximo problema considerado foi o de adaptação de enlace por meio do *bit loading*, que é e versão discreta do algoritmo de *water-filling*, este a solução clássica para o problema de alocação de potência em canais seletivos em frequência e, em particular, sistemas multiportadora. Como contribuição, tem-se a formulação do problema de *bit loading* como um problema de aprendizado por reforço. Foi tratada a modelagem necessária, o que inclui a definição de estados, recompensas e ações necessárias ao agente inteligente para que suas decisões possam ser guiadas, de forma a maximizar um valor de recompensa acumulado. Resultados de simulação mostraram que a abordagem proposta é capaz de aproximar a solução ótima para o problema, dada pelo algoritmo de Levin-Campello, porém sem a necessidade de um tratamento analítico para o problema por meio da formulação de *gap* de capacidade.

Finalmente, abordou-se o problema de seleção e escalonamento de usuários em um sistema de comunicação multiusuário. Inicialmente, foram feitas considerações sobre os diferentes tipos de aplicações que trafegam nos atuais sistemas de comunicação e como essa heterogeneidade gera diferentes requisitos de qualidade de serviço a serem atendidos, estes em geral conflitantes. Logo, é introduzida a necessidade de se considerar o escalonamento como um problema multi-objetivo. Como contribuição, apresentou-se uma proposta de algoritmo de seleção e escalonamento de usuários, denominado CH-RL, baseado em aprendizado por reforço multiobjetivo e utilizando-se da iteração no fecho convexo como forma de resolver o problema de aprendizagem. O algoritmo analisa o problema de escalonamento em três dimensões (vazão, atraso de escalonamento e preenchimento do *buffer* do usuário) e em um horizonte de tomada de decisão de um quadro de rádio, selecionando as melhores oportunidades de transmissão utilizando o conceito de fecho convexo e resultados fornecidos por algoritmos de predição de canal e de tráfego, este implementados por meio de filtros adaptativos que utilizam o algoritmo *set-membership affine projection*. Resultados de simulação mostraram que a abordagem proposta apresenta resultados superiores em termos de perda de pacotes e atraso na entrega de pacotes quando comparado à abordagem clássica da literatura, dada pelo algoritmo M-LWDF, para lidar com o escalonamento de dados sensíveis ao atraso, como é o caso da transmissão de vídeo em tempo real.

6.1 PROPOSTAS DE TRABALHOS FUTUROS

Este trabalho identificou potencias ganhos que podem ser obtidos em sistemas de comunicação por meio das técnicas de aprendizado de máquina (e, em especial, do aprendizado por reforço). Diversos aspectos e frentes de trabalho podem ser identificados e aprofundados. Algumas sugestões para a extensão deste trabalho são:

- a utilização de técnicas de aprendizado de máquina e aprendizado por reforço em redes do tipo SON (*self-organizing network*), que possui como objetivo tornar mais simples o planejamento, a configuração e o gerenciamento de redes de acesso. A própria estrutura e organização dessas redes sugere a utilização de mecanismos de aprendizado para sua auto-configuração;
- a formulação e solução dos problemas de adaptação de enlace utilizando aprendizado por reforço considerando a utilização de múltiplas antenas na camada física;
- considerar técnicas de redução de dimensionalidade e aceleração de convergência os algoritmos propostos e apresentados;
- a abordagem do problema de escalonamento no enlace reverso (*uplink*) e a otimização conjunta de seleção de usuários e alocação não uniforme de potência entre os recursos de rádio;

- outras formas de reconfigurabilidade dos rádios, que não leve apenas em consideração eficiência espectral, mas também eficiência na utilização de potência e diminuição dos níveis de interferência, sobretudo no cenário cognitivo;
- modificação do algoritmo de forma a simular estratégias de alocação de recursos so tipo persistentes, como uma forma de diminuir o overhead na transmissão dos quadros de rádio;
- considerar um número maior de dimensões a serem analisadas pelo problema de seleção e escalonamento de usuários, como forma de contemplar outras aplicações, tais como aplicações médias, sinalização, jogos em tempo real, telemetria etc.
- a utilização de aprendizado por reforço e aprendizado de máquina em outras camadas de comunicação além da camada física, como as camadas de acesso, rede e aplicação.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] DANIELS, R. C. *Machine Learning for Link Adaptation in Wireless Networks*. Tese (Doutorado) — THE UNIVERSITY OF TEXAS AT AUSTIN, Texas, USA, dez. 2011.

[2] MITOLA III, J. Cognitive radio for flexible mobile multimedia communications. *Mob. Netw. Appl.*, Kluwer Academic Publishers, Hingham, MA, USA, v. 6, n. 5, p. 435–441, set. 2001.

[3] HAYKIN, S. Cognitive radio: brain-empowered wireless communications. *Selected Areas in Communications, IEEE Journal on*, v. 23, n. 2, p. 201–220, 2005.

[4] AKYILDIZ, I. et al. A survey on spectrum management in cognitive radio networks. *Communications Magazine, IEEE*, v. 46, n. 4, p. 40–48, 2008.

[5] SHIN, K. et al. Cognitive radios for dynamic spectrum access: from concept to reality. *Wireless Communications, IEEE*, v. 17, n. 6, p. 64–74, 2010.

[6] GANDETTO, M.; REGAZZONI, C. Spectrum sensing: A distributed approach for cognitive terminals. *Selected Areas in Communications, IEEE Journal on*, v. 25, n. 3, p. 546–557, 2007.

[7] DEEPA, B.; IYER, A.; MURTHY, C. Cyclostationary-based architectures for spectrum sensing in ieee 802.22 wran. In: *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. [S.l.: s.n.], 2010. p. 1–5.

[8] KIM, Y. M. et al. An alternative energy detection using sliding window for cognitive radio system. In: *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on*. [S.l.: s.n.], 2008. v. 1, p. 481–485.

[9] MA, J.; LI, G.; JUANG, B.-H. Signal processing in cognitive radio. *Proceedings of the IEEE*, v. 97, n. 5, p. 805–823, 2009.

[10] GRACE, D. Cognitive communications; the intelligent future? In: *Wireless Mobile and Computing (CCWMC 2009), IET International Communication Conference on*. [S.l.: s.n.], 2009. p. xxvi–xxvi.

[11] BKASSINY, M. et al. Wideband spectrum sensing and non-parametric signal classification for autonomous self-learning cognitive radios. *Wireless Communications, IEEE Transactions on*, v. 11, n. 7, p. 2596–2605, 2012.

[12] MODY, A. et al. Machine learning based cognitive communications in white as well as the gray space. In: *Military Communications Conference, 2007. MILCOM 2007. IEEE*. [S.l.: s.n.], 2007. p. 1–7.

[13] CLANCY, C. et al. Applications of machine learning to cognitive radio networks. *Wireless Communications, IEEE*, v. 14, n. 4, p. 47–52, 2007.

[14] RONDEAU, T. W.; BOSTIAN, C. W. *Artificial Intelligence in Wireless Communications*. Norwood, MA: Artech House, 2009.

[15] BALDO, N.; ZORZI, M. Learning and adaptation in cognitive radios using neural networks. In: *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*. [S.l.: s.n.], 2008. p. 998–1003.

[16] BALDO, N. et al. A neural network based cognitive controller for dynamic channel selection. In: *Communications, 2009. ICC '09. IEEE International Conference on*. [S.l.: s.n.], 2009. p. 1–5.

[17] TANG, Y.-J.; ZHANG, Q. yu; LIN, W. Artificial neural network based spectrum sensing method for cognitive radio. In: *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*. [S.l.: s.n.], 2010. p. 1–4.

[18] TAJ, M. I.; AKIL, M. Cognitive radio spectrum e volution prediction using a rtificial neural networks based multivariate t ime series modelling. In: *Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European*. [S.l.: s.n.], 2011. p. 1–6.

[19] POPOOLA, J.; OLST, R. van. A novel modulation-sensing method. *Vehicular Technology Magazine, IEEE*, v. 6, n. 3, p. 60–69, 2011.

[20] HU, H.; SONG, J.; WANG, Y. Signal classification based on spectral correlation analysis and svm in cognitive radio. In: *Advanced Information Networking and Applications, 2008. AINA 2008. 22nd International Conference on*. [S.l.: s.n.], 2008. p. 883–887.

[21] XU, G.; LU, Y. Channel and modulation selection based on support vector machines for cognitive radio. In: *Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006.International Conference on*. [S.l.: s.n.], 2006. p. 1–4.

[22] HAI-YUAN, L.; SUN, J.-C. A modulation type recognition method using wavelet support vector machines. In: *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*. [S.l.: s.n.], 2009. p. 1–4.

[23] ZHAO, Q.; TONG, L.; SWAMI, A. Decentralized cognitive mac for dynamic spectrum access. In: *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*. [S.l.: s.n.], 2005. p. 224–232.

[24] HAKIM, K. et al. Efficient dynamic spectrum sharing in cognitive radio networks: Centralized dynamic spectrum leasing (c-dsl). *Wireless Communications, IEEE Transactions on*, v. 9, n. 9, p. 2956–2967, 2010.

[25] LATIFA, B.; GAO, Z.; LIU, S. No-regret learning for simultaneous power control and channel allocation in cognitive radio networks. In: *Computing, Communications and Applications Conference (ComComAp), 2012*. [S.l.: s.n.], 2012. p. 267–271.

[26] ZHU, Q.; HAN, Z.; BASAR, T. No-regret learning in collaborative spectrum sensing with malicious nodes. In: *Communications (ICC), 2010 IEEE International Conference on*. [S.l.: s.n.], 2010. p. 1–6.

[27] GEIRHOFER, S.; TONG, L.; SADLER, B. M. Cognitive Medium Access: A Protocol for Enhancing Coexistence in WLAN Bands. In: *Proceedings of IEEE Global Telecommunications Conference, GLOBECOM '07*. Washington, DC: [s.n.], 2007. p. 3558–3562.

[28] ZHAO, Q. et al. Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: A POMDP framework. v. 25, n. 3, p. 589–600, mar. 2007.

[29] COUPECHOUX, M.; KELIF, J. M.; GODLEWSKI, P. SMDP approach for JRRM analysis in heterogeneous networks. In: *Proceedings of 14th European Wireless Conference, EW 2008.* Prage, Czech Republic: [s.n.], 2008. p. 1–7.

[30] GALINDO-SERRANO, A.; GIUPPONI, L. Distributed Q-Learning for Interference Control in OFDMA-Based Femtocell Networks. In: *IEEE 71st Vehicular Technology Conference, VTC 2010-Spring.* Taipei: [s.n.], 2010. p. 1–5.

[31] GOLDSMITH, A. *Wireless Communications*. New York, NY: Cambridge University Press, 2005.

[32] KANT, S.; JENSEN, T. L. *Fast link adaptation for IEEE 802.11n*. Dissertação (M. S. thesis) — Aalborg University, Denmark, fev. 2007.

[33] WANG, C. et al. On the performance of the mimo zero-forcing receiver in the presence of channel estimation error. *Wireless Communications, IEEE Transactions on*, v. 6, n. 3, p. 805–810, 2007.

[34] SUNG, C.-K. et al. Adaptive bit-interleaved coded ofdm with reduced feedback information. *Communications, IEEE Transactions on*, v. 55, n. 9, p. 1649–1655, 2007.

[35] MANDKE, K. et al. Early results on hydra: A flexible mac/phy multihop testbed. In: *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*. [S.l.: s.n.], 2007. p. 1896–1900.

[36] TIAN, C.; YUAN, D. Cross layer opportunistic scheduling for multiclass users in cognitive radio networks. In: *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*. [S.l.: s.n.], 2008. p. 1–4.

[37] UDGATA, S.; KUMAR, K.; SABAT, S. Swarm intelligence based resource allocation algorithm for cognitive radio network. In: *Parallel Distributed and Grid Computing (PDGC), 2010 1st International Conference on*. [S.l.: s.n.], 2010. p. 324–329.

[38] ZHANG, L. et al. Proportional fair scheduling based on primary user traffic patterns for spectrum sensing in cognitive radio networks. In: *Communications in China (ICCC), 2012 1st IEEE International Conference on*. [S.l.: s.n.], 2012. p. 302–306.

[39] CAPOZZI, F. et al. Downlink packet scheduling in lte cellular networks: Key design issues and a survey. *Communications Surveys Tutorials, IEEE*, v. 15, n. 2, p. 678–700, 2013.

[40] SESIA, S. *LTE - The UMTS Long Term Evolution: From Theory to Practice*. Hoboken, New Jersey: Wiley-Interscience, 2012.

[41] ZHANG, H.; PRASAD, N.; RANGARAJAN, S. Mimo downlink scheduling in lte systems. In: *INFOCOM, 2012 Proceedings IEEE*. [S.l.: s.n.], 2012. p. 2936–2940.

[42] CHUNG, Y.-L.; JANG, L.-J.; TSAI, Z. An efficient downlink packet scheduling algorithm in lte-advanced systems with carrier aggregation. In: *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*. [S.l.: s.n.], 2011. p. 632–636.

[43] BKASSINY, M.; LI, Y.; JAYAWEERA, S. A survey on machine-learning techniques in cognitive radios. *Communications Surveys Tutorials, IEEE*, v. 15, n. 3, p. 1136–1159, Third 2013.

[44] PUTERMAN, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, New Jersey: Wiley-Interscience, 2005.

[45] LAGOUDAKIS, M. G.; PARR, R. Least-Squares Policy Iteration. *Journal of Machine Learning Research*, v. 4, p. 1107–1149, dez. 2003.

[46] BERTSEKAS, D. P. *Dynamic Programming and Optimal Control - Vol. II*. 3. ed. Cambridge, MA: Athena Scientific, 2007.

[47] SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[48] BAUMEISTER, J. *Introdução a Teoria de Controle e Programação Dinâmica*. Rio de Janeiro, RJ: IMPA (Projeto Euclides), 2008.

[49] BERTSEKAS, D. P. *Dynamic Programming and Optimal Control - Vol. I*. 3. ed. Cambridge, MA: Athena Scientific, 2007.

[50] PLASS, S. et al. *Multi-Carrier Spread Spectrum 2007: Proceedings from the 6th International Workshop on Multi-Carrier Spread Spectrum*. Herrsching, Germany: Springer, 2007.

[51] SZEPESVARI, C. *Algorithms for Reinforcement Learning*. [S.l.]: Morgan and Claypool Publishers, 2010.

[52] BUSONIU, L. et al. *Reinforcement Learning and Dynamic Programming Using Function Approximators*. Boca Raton, FL: CRC Press, 2010.

[53] BARRETT, L.; NARAYANAN, S. Learning all optimal policies with multiple criteria. In: *Proceedings of the 25th international conference on Machine learning*. New York, NY, USA: ACM, 2008. p. 41–47.

[54] VAMPLEW, P. et al. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Mach. Learn.*, Kluwer Academic Publishers, v. 84, n. 1-2, p. 51–80, jul. 2011. ISSN 0885-6125.

[55] MANNOR, S.; SHIMKIN, N. The steering approach for multi-criteria reinforcement learning. In: *Learning Advances in Neural Information Processing Systems*. Vancouver, Canada: [s.n.], 2001. p. 1563–1570.

[56] MANNOR, S.; SHIMKIN, N.; MAHADEVAN, S. A geometric approach to multi-criterion reinforcement learning. *Journal of Machine Learning Research*, v. 5, p. 360, 2004.

[57] GABOR, Z.; KALMAR, Z.; SZEPESVARI, C. Multi-criteria reinforcement learning. In: *The fifteenth international conference on machine learning*. [S.l.: s.n.], 1998.

[58] Joseph O'Rourke. *Computational Geometry in C.* : Cambridge University Press, 1998.

[59] GOECKEL, D. Adaptive coding for time-varying channels using outdated fading estimates. *Communications, IEEE Transactions on*, v. 47, n. 6, p. 844–855, 1999.

[60] Goldsmith, A.J. and Soon-Ghee Chua. Adaptive coded modulation for fading channels. *Communications, IEEE Transactions on*, v. 46, n. 5, p. 595–602, 1998.

[61] GOLDSMITH, A.; CHUA, S.-G. Variable-rate variable-power MQAM for fading channels. *Communications, IEEE Transactions on*, v. 45, n. 10, p. 1218–1230, 1997.

[62] HOLE, K.; HOLM, H.; OIEN, G. Adaptive multidimensional coded modulation over flat fading channels. *Selected Areas in Communications, IEEE Journal on*, v. 18, n. 7, p. 1153–1158, 2000.

[63] PURSLEY, M.; SHEA, J. Adaptive nonuniform phase-shift-key modulation for multimedia traffic in wireless networks. *Selected Areas in Communications, IEEE Journal on*, v. 18, n. 8, p. 1394–1407, 2000.

[64] Recommendation ITU-R M.1645. *Framework and overall objectives of the future development of IMT-2000 and systems beyond IMT-2000*. [S.l.], 2003.

[65] JENSEN, T. et al. Fast link adaptation for mimo ofdm. *Vehicular Technology, IEEE Transactions on*, v. 59, n. 8, p. 3766–3778, 2010.

[66] DANIELS, R. C.; CARAMANIS, C. M.; HEATH, J. R. W. A Supervised Learning Approach to Adaptation in Practical MIMO-OFDM Wireless Systems. In: *Proc. of IEEE Global Telecommunications Conference, 2008 - GLOBECOM 2008*. New Orleans, LO: [s.n.], 2008. p. 1–5.

[67] YANG, S. C. *OFDMA System Analysis and Design*. Norwood, MA: Artech House, 2010.

[68] DANIELS, R. C.; CARAMANIS, C. M.; HEATH, J. R. W. Adaptation in Convolutionally Coded MIMO-OFDM Wireless Systems Through Supervised Learning and SNR Ordering. v. 59, n. 1, p. 114–126, jan. 2010.

[69] DANIELS, R. C.; CARAMANIS, C. M.; HEATH, J. R. W. Online Adaptive Modulation and Coding with Support Vector Machines. In: *Proc. of 2010 European Wireless Conference (EW)*. Lucca, Italy: [s.n.], 2010. p. 718–724.

[70] YIGIT, H.; KAVAK, A. Adaptation using Neural Network in Frequency Selective MIMO-OFDM Systems. In: *Proc. of 5th IEEE International Symposium on Wireless Pervasive Computing (ISWPC), 2010*. Modena, Italy: [s.n.], 2010. p. 390–394.

[71] ALPAYDIN, E. *Introduction to Machine Learning*. 2. ed. Cambridge, MA: MIT Press, 2010.

[72] SCHENK, T. *RF Imperfections in High-rate Wireless Systems: Impact and Digital Compensation*. Dordrecht, the Netherlands: Springer, 2008.

[73] HONG, X.; WANG, C.-X.; THOMPSON, J. Interference Modeling of Cognitive Radio Networks. In: *Vehicular Technology Conference, IEEE VTC Spring 2008*. Marina Bay, Singapore: [s.n.], 2008. p. 1851–1855.

[74] ALJUAID, M.; YANIKOMEROGLU, H. Investigating the Gaussian Convergence of the Distribution of the Aggregate Interference Power in Large Wireless Networks. v. 59, n. 9, p. 4418–4424, nov. 2010.

[75] LIN, J.; STEFANOV, A. Exact pairwise error probability for block fading ODFM systems. In: *Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE*. [S.l.: s.n.], 2006. v. 2, p. 1120–1124.

[76] MINN, H.; ZENG, M.; BHARGAVA, V. On ARQ scheme with adaptive error control. *Vehicular Technology, IEEE Transactions on*, v. 50, n. 6, p. 1426–1436, 2001.

[77] HALEEM, M. A.; CHANDRAMOULI, R. Adaptive Stochastic Iterative Rate Selection for Wireless Channels. v. 8, n. 5, p. 292–294, maio 2004.

[78] MISRA, A.; KRISHNAMURTHY, V.; SCHOBER, R. Stochastic Learning Algorithms for Adaptive Modulation. In: *Proc. of IEEE 6th Workshop on Signal Processing Advances in Wireless Communications - SPAWC 2005*. New York, NY: [s.n.], 2005. p. 756–760.

[79] HAYKIN, S. *Redes Neurais: Principios e Pratica*. 2. ed. São Paulo, SP: Bookman, 2001.

[80] SMART, W. D. *Making Reinforcement Learning Work on Real Robots*. Tese (Doutorado) — Brown University, Providence, Rhode Island, maio 2002.

[81] RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3. ed. Upper Saddle River, New Jersey: Prentice Hall, 2009.

[82] 3GPP. *3GPP TR 25.996 V8.5.0 - Spatial Channel Model for Multiple Input Multiple Output (MIMO) Simulations (Release 6)*. [S.l.], set. 2003.

[83] SALO, J. et al. *MATLAB implementation of the 3GPP Spatial Channel Model (3GPP TR 25.996)*. jan. 2005. Disponível em: <http://www.tkk.fi/Units/Radio/scm/>.

[84] LAMARCA, M.; REY, F. Indicators for PER Prediction in Wireless Systems: A Comparative Study. In: *Vehicular Technology Conference, VTC Spring*. Stockholm, Sweden: [s.n.], 2005. v. 2, n. 30, p. 792–796.

[85] PENG, F.; ZHANG, J. Adaptive Modulation and Coding for IEEE 802.11n. In: *IEEE Wireless Communications and Networking Conference*. Kowloon, Hong kong: [s.n.], 2007. p. 656–661.

[86] CSáJI, B. C.; MONOSTORI, L. Value Function Based Reinforcement Learning in Changing Markovian Environments. *Journal of Machine Learning Research*, v. 9, p. 1679–1709, jun. 2008.

[87] FETTWEIS, G. et al. Dirty RF: A New Paradigm. *International Journal of Wireless Information Networks*, v. 14, n. 2, p. 133–148, jun. 2007.

[88] MAHMOOD, A. *Computationally Efficient Adaptive Algorithms for Multicarrier Physical Layer*. Tese (Doutorado) — Ecole Nationale Superieure des Telecommunications, Paris, France, 2008.

[89] COVER, T. M.; THOMAS, J. A. *Elements of Information Theory*. Hoboken, New Jersey: Wiley-Interscience, 2006.

[90] GARCIA-ARMADA, A. Snr gap approximation for m-psk-based bit loading. *Wireless Communications, IEEE Transactions on*, v. 5, n. 1, p. 57–60, 2006.

[91] KRONGOLD, B. S.; RAMCHANDRAN, K.; JONES, D. Computationally efficient optimal power allocation algorithm for multicarrier communication systems. In: *Communications,*

1998. ICC 98. Conference Record. 1998 IEEE International Conference on. [S.l.: s.n.], 1998. v. 2, p. 1018–1022.

[92] SONALKAR, R.; SHIVELY, R. An efficient bit-loading algorithm for dmt applications. *Communications Letters, IEEE*, v. 4, n. 3, p. 80–82, 2000.

[93] PAPANDREOU, N.; ANTONAKOPOULOS, T. A new computationally efficient discrete bit-loading algorithm for dmt applications. *Communications, IEEE Transactions on*, v. 53, n. 5, p. 785–789, 2005.

[94] CHOW, P.; CIOFFI, J.; BINGHAM, J. A. C. A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels. *Communications, IEEE Transactions on*, v. 43, n. 234, p. 773–775, 1995.

[95] CZYLWIK, A. Adaptive ofdm for wideband radio channels. In: *Global Telecommunications Conference, 1996. GLOBECOM '96. 'Communications: The Key to Global Prosperity*. [S.l.: s.n.], 1996. v. 1, p. 713–718 vol.1.

[96] FISCHER, R. F. H.; HUBER, J. A new loading algorithm for discrete multitone transmission. In: *Global Telecommunications Conference, 1996. GLOBECOM '96. 'Communications: The Key to Global Prosperity*. [S.l.: s.n.], 1996. v. 1, p. 724–728 vol.1.

[97] BACCARELLI, E.; FASANO, A.; BIAGI, M. Novel efficient bit-loading algorithms for peak-energy-limited adsl-type multicarrier systems. *Signal Processing, IEEE Transactions on*, v. 50, n. 5, p. 1237–1247, 2002.

[98] MAHMOOD, A.; JAFFROT, E. Wlcp1-07: An efficient methodology for optimal discrete bit-loading with spectral mask constraints. In: *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE*. [S.l.: s.n.], 2006. p. 1–5.

[99] MAHMOOD, A.; BELFIORE, J. C. Improved 3-db subgroup based algorithm for optimal discrete bit-loading. In: *Sarnoff Symposium, 2008 IEEE*. [S.l.: s.n.], 2008. p. 1–5.

[100] WILLINK, T.; WITTKE, P. Optimization and performance evaluation of multicarrier transmission. *Information Theory, IEEE Transactions on*, v. 43, n. 2, p. 426–440, 1997.

[101] PALOMAR, D.; FONOLLOSA, J. Practical algorithms for a family of waterfilling solutions. *Signal Processing, IEEE Transactions on*, v. 53, n. 2, p. 686–695, 2005.

[102] Bartolome, D. and PeÌĄrez-Neira, A.I. Practical implementation of bit loading schemes for multiantenna multiuser wireless ofdm systems. *Communications, IEEE Transactions on*, v. 55, n. 8, p. 1577–1587, 2007.

[103] LEE, J. *Power Allocation for Multi-user Multi-carrier Communication Systems*. Tese (Doutorado) — Stanford University, California, mar. 2003.

[104] LI, Y.; RYAN, W. Mutual-information-based adaptive bit-loading algorithms for ldpc-coded ofdm. *Wireless Communications, IEEE Transactions on*, v. 6, n. 5, p. 1670–1680, 2007.

[105] CAMPELLO, J. Practical bit loading for dmt. In: *Communications, 1999. ICC '99. 1999 IEEE International Conference on*. [S.l.: s.n.], 1999. v. 2, p. 801–805 vol.2.

[106] CAMPELLO, J. Optimal discrete bit loading for multicarrier modulation systems. In: *Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on*. [S.l.: s.n.], 1998. p. 193–.

[107] LOZANO, A.; JINDAL, N. Transmit diversity vs. spatial multiplexing in modern mimo systems. *Wireless Communications, IEEE Transactions on*, v. 9, n. 1, p. 186–197, 2010.

[108] RUSEK, F.; LOZANO, A.; JINDAL, N. Mutual information of iid complex gaussian signals on block rayleigh-faded channels. *Information Theory, IEEE Transactions on*, v. 58, n. 1, p. 331–340, 2012.

[109] MATAS, D.; LAMARCA, M. Optimum power allocation and bit loading with code rate constraints. In: *Signal Processing Advances in Wireless Communications, 2009. SPAWC '09. IEEE 10th Workshop on*. [S.l.: s.n.], 2009. p. 687–691.

[110] LOZANO, A.; TULINO, A.; VERDU, S. Optimum power allocation for parallel gaussian channels with arbitrary input distributions. *Information Theory, IEEE Transactions on*, v. 52, n. 7, p. 3033–3051, 2006.

[111] COX, C. *An Introduction to LTE*. Hoboken, New Jersey: Wiley-Interscience, 2012.

[112] FORUM, W. *WiMAX System Evaluation Methodology (EMD)*. jul. 2008. Disponível em: <http://www.cse.wustl.edu/ jain/wimax/ftp/wimax$systemevaluationmethodologyv21.pdf$>.

[113] CORVINO, V.; TRALLI, V.; VERDONE, R. Cross-layer radio resource allocation for multicarrier air interfaces in multicell multiuser environments. *Vehicular Technology, IEEE Transactions on*, v. 58, n. 4, p. 1864–1875, 2009.

[114] CHOI, Y.-J.; BAHK, S. Waf: wireless-adaptive fair scheduling for multimedia stream in time division multiplexed packet cellular systems. In: *Computers and Communication, 2003. (ISCC 2003). Proceedings. Eighth IEEE International Symposium on*. [S.l.: s.n.], 2003. p. 1085–1090 vol.2.

[115] AZGIN, A.; KRUNZ, M. Scheduling in wireless cellular networks under probabilistic channel information. In: *Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on*. [S.l.: s.n.], 2003. p. 89–94.

[116] MORELL, A. et al. Robust scheduling in mimo-ofdm multi-user systems based on convex optimization. In: *Computational Advances in Multi-Sensor Adaptive Processing, 2005 1st IEEE International Workshop on*. [S.l.: s.n.], 2005. p. 113–116.

[117] YU, Y.; GIANNAKIS, G. Joint congestion control and ofdma scheduling for hybrid wireline-wireless networks. In: *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*. [S.l.: s.n.], 2007. p. 973–981.

[118] HANG, J.; FAN, Z.; SHE, F. Performance analysis of power optimization and user scheduling in multi-user mimo-ofdm systems. In: *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*. [S.l.: s.n.], 2009. v. 3, p. 238–242.

[119] RAMLI, H. et al. Performance of well known packet scheduling algorithms in the downlink 3gpp lte system. In: *Communications (MICC), 2009 IEEE 9th Malaysia International Conference on*. [S.l.: s.n.], 2009. p. 815–820.

[120] ITURRALDE, M. et al. Resource allocation using shapley value in lte networks. In: *Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on*. [S.l.: s.n.], 2011. p. 31–35.

[121] LEITE, J. P.; VIEIRA, R. D.; CARVALHO, P. H. P. de. OFDM Channel Prediction Using Set-Membership Affine Projection Algorithm in Time-Varying Wireless Channel. In: *IEEE 10th Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2009) (July 2009)*. [S.l.: s.n.], 2009.

[122] LEITE, J. P.; VIEIRA, R. D.; CARVALHO, P. H. P. de. PrediÃğÃčo de Canal em Sistemas OFDM Utilizando o Algoritmo Set-Membership Affine Projection. In: *XXVII Simposio Brasileiro de Telecomunicacoes - SBrT 2009*. [S.l.: s.n.], 2009.

[123] DINIZ, P. S. R. *Adaptive Filtering: Algorithms and Practical Implementation*. [S.l.]: Kluwer Academic Publishers, 2002.

[124] PIRO, G. et al. Two-level downlink scheduling for real-time multimedia services in lte networks. *Multimedia, IEEE Transactions on*, v. 13, n. 5, p. 1052–1065, Oct 2011.

[125] SAFA, H.; TOHME, K. Lte uplink scheduling algorithms: Performance and challenges. In: *Telecommunications (ICT), 2012 19th International Conference on*. [S.l.: s.n.], 2012. p. 1–6.

[126] KIM, A.; PARK, C.; JEONG, S.-H. Performance evaluation of downlink scheduling algorithms for video/voice transport over wireless networks. In: *Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on*. [S.l.: s.n.], 2013. p. 791–794.

[127] ANDREWS, M. et al. Providing quality of service over a shared wireless link. *Communications Magazine, IEEE*, v. 39, n. 2, p. 150–154, Feb 2001.

[128] JAIN, R.; DAH-MING, W.; HAWE, W. R. A quantitive measure of fairness and discrimination for resource allocation in shared computer systems. *Computer Systems, ACM Transactions on*, 1984.

ANEXOS

I. LONG TERM EVOLUTION

O LTE é uma proposta de evolução para o UMTS (*Universal Mobile Telecommunications System*), sistema de telefonia celular de terceira geração. Elaborado e discutido por uma parceria denominada 3GPP (*Third Generation Partnership Project*), o LTE busca manter a competitividade da linha evolutiva que advém do sistema GSM (*Global System for Mobile Communications*).

Neste anexo, será considerada uma breve descrição da camada física do enlace direto do padrão LTE operando em FDD para uma largura de banda de 10 MHz e prefixo cíclico curto, que corresponde ao cenário utilizado para realizar as simulações apresentadas.

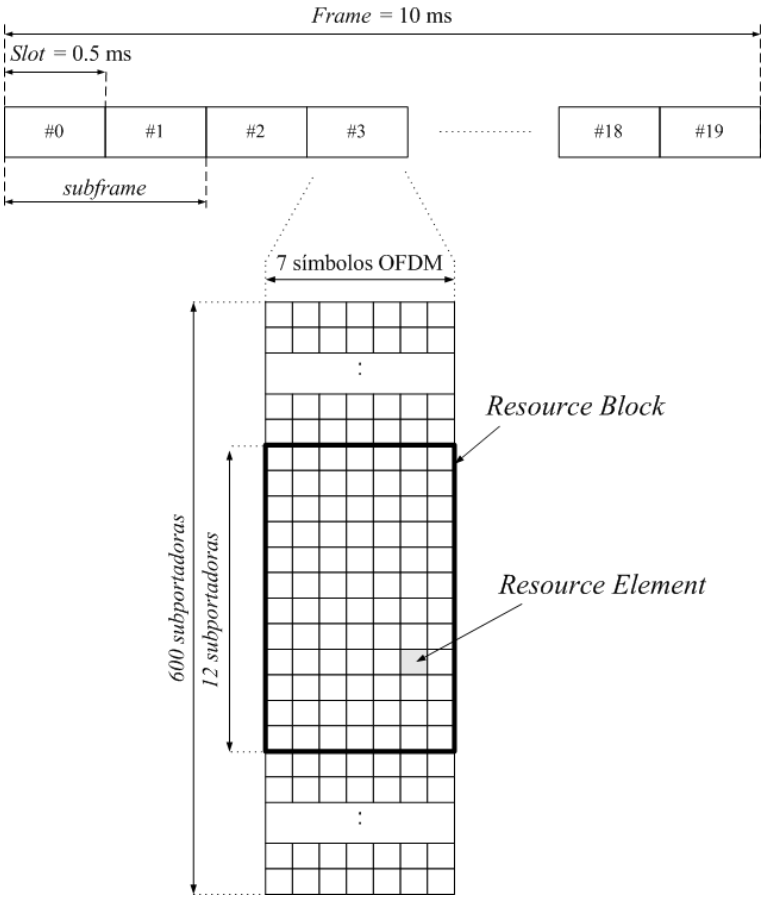


Figura I.1: Estrutura de quadro no enlace direto do padrão LTE para uma largura de banda de 10 MHz.

A técnica de múltiplo acesso utilizada no enlace direto do sistema LTE é o OFDMA. Como consequência, o quadro de rádio (*frame*) possui dimensões tanto no domínio do tempo como no domínio da frequência, conforme mostrado na Fig. I.1. O quadro de rádio tem duração de 10 ms e é composto por 10 subquadros, cada um com 1 ms de duração. Cada subquadro, por sua vez, é

constituído de 2 *slots*, cada um com duração de 0,5 ms.

Logo, pela descrição apresentada no parágrafo anterior, o quadro de rádio é formado por 20 *slots*, numerados de 0 a 19, conforme mostrado na Fig. I.1. Cada *slot* contém de 7 símbolos OFDM. Para a banda de transmissão de 10 MHz, são utilizadas 600 subportadoras, com espaçamento de 15 kHz entre elas.

A menor unidade de recurso recebe a denominação de elemento de recurso (*resource element*), e corresponde a uma subportadora dentro de um símbolo OFDM. O bloco de recursos (*resource block*) é definido como o conjunto de 7 símbolos OFDM contíguos no domínio do tempo e de 12 subportadoras adjacentes no domínio da frequência, e consiste na menor unidade de recursos que pode ser alocada para um usuário.

A distribuição das subportadoras que carregam símbolos pilotos, utilizados para a estimação de canal, é mostrada na Fig. I.2 para o caso em que a estação rádio-base opera utilizando quatro antenas de transmissão. Pode-se perceber que há símbolos de referência no primeiro e no quarto símbolo OFDM de um *slot*. Existe um espaçamento de seis subportadoras entre os símbolos de referência, sendo que os símbolos de referência localizados no quarto símbolo OFDM de um *slot* são alocados com um *offset* de quatro subportadoras com relação ao primeiro símbolo do *slot*.

Mais especificamente, no padrão LTE, a estação rádio-base pode possuir uma, duas ou quatro antenas, e quando duas ou mais antenas são utilizadas para transmissão, os símbolos de referência são posicionados de tal forma que sejam ortogonais, de forma a não interferir uns nos outros. Essa ortogonalidade é obtida garantindo que nenhum sinal seja transmitido nos elementos de recurso que são utilizados por uma determinada antena para a transmissão de símbolos de referência. Estes são marcados na Fig. I.2 como as portadoras não utilizadas.

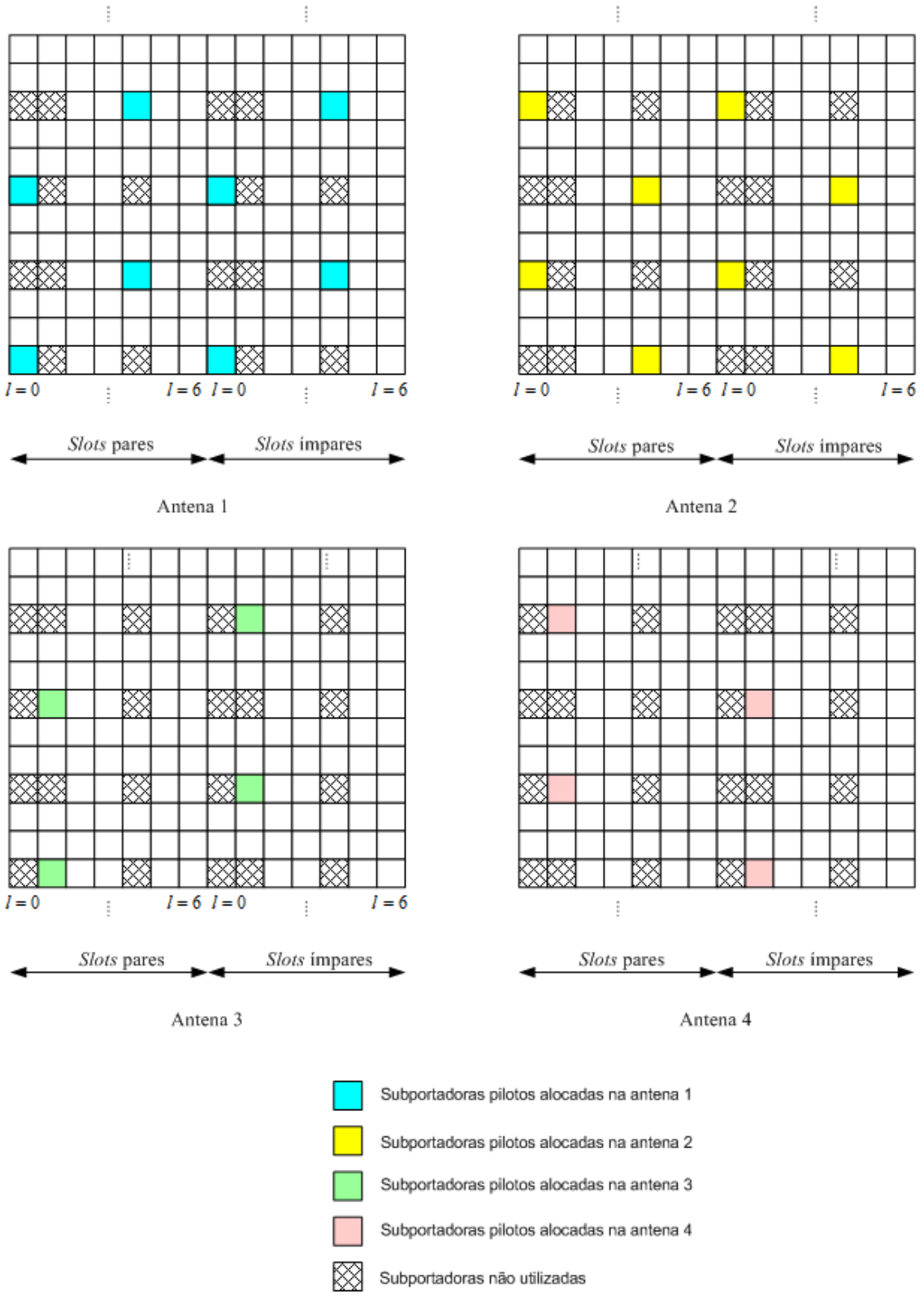


Figura I.2: Alocação de subportadoras que carregam símbolos pilotos no enlace direto do padrão LTE para quatro antenas de transmissão.

II. MODELO DE CANAL SCM

Este anexo apresenta uma breve descrição do modelo espacial de canal do 3GPP, SCM, utilizado para simulação de sistemas MIMO.

O modelo foi desenvolvido para que as técnicas MIMO propostas para os sistemas de terceira geração (3G) pudessem ser comparadas e comparadas em um ambiente *outdoor*. Ele é largamente utilizado para modelagem de canal de ambientes urbanos microcelulares e macrocelulares.

O SCM é um modelo empírico baseado em considerações físicas sobre os espalhadores (*scatterers*), e especifica três ambientes de propagação: macrocélula suburbana, macrocélula urbana e microcélula urbana. O sinal recebido pelo terminal móvel consiste em N versões atrasadas do sinal enviado. Os N multipercursos são caracterizados de acordo com o tipo de ambiente escolhido, e seus valores variam de 6 a 20 multipercursos.

Cada componente de multipercurso corresponde a um *cluster* de M subcaminhos (*subpaths*), cada um caracterizando um determinado espalhador. Os M subcaminhos definem um *cluster* de espalhadores que possuem o mesmo atraso de multipercurso, mas suas amplitudes e fases (os ângulos de partida e chegada) são variáveis aleatórias, produzindo desvanecimento do tipo Rayleigh ou Rice.

Matematicamente, $h_{u,s,n}(t)$, o n-ésimo componente de multipercurso da u-ésima antena de transmissão para a s-ésima antena de recepção, é dado por:

$$h_{u,s,n}(t) = \sqrt{\frac{P_n \sigma_s}{M}} \sum_{m=1}^{M} \begin{pmatrix} \sqrt{G_{BS}\left(\theta_{n,m,AoD}\right)} \exp\left(j\left[kd_s \sin\left(\theta_{n,m,AoD} + \Phi_{n,m}\right)\right]\right) \times \\ \sqrt{G_{MS}\left(\theta_{n,m,AoA}\right)} \exp\left(jkd_u \sin\left(\theta_{n,m,AoA}\right)\right) \times \\ \exp\left(jk \left\|\mathbf{v}\right\| \cos\left(\theta_{n,m,AoA} - \theta_v\right) t\right) \end{pmatrix}, \quad \text{(II.1)}$$

em que são definidos os seguintes parâmetros, ilustrados na Fig. II.1:

- P_n é a potência do n-ésimo multipercurso;
- σ_s é a desvio padrão da componente de sombreamento (*shadowing*);
- $\theta_{n,m,AoD}$ é o ângulo de partida do m-ésimo *subpath* do n-ésimo multipercurso;
- $\theta_{n,m,AoA}$ é o ângulo de chegada do m-ésimo *subpath* do n-ésimo multipercurso;
- $G_{BS}\left(\theta_{n,m,AoD}\right)$ é o ganho de cada elemento do conjunto de antenas na estação rádio-base;
- $G_{MS}\left(\theta_{n,m,AoA}\right)$ é o ganho de cada elemento do conjunto de antenas no terminal móvel;
- k é o número de onda, dado por $\frac{2\pi}{\lambda}$, em que λ é o comprimento de onda;

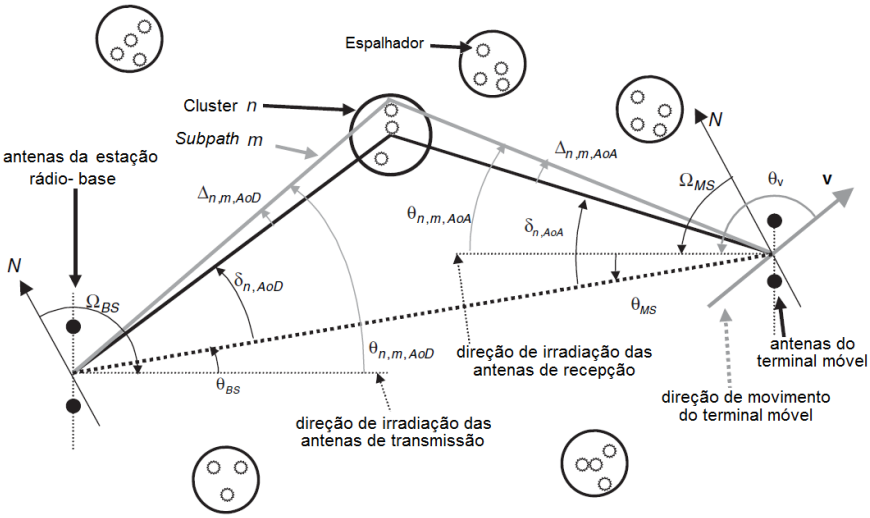
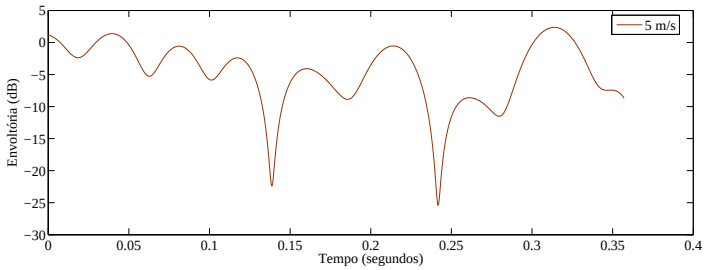


Figura II.1: Modelo espacial do 3GPP para simulações MIMO.

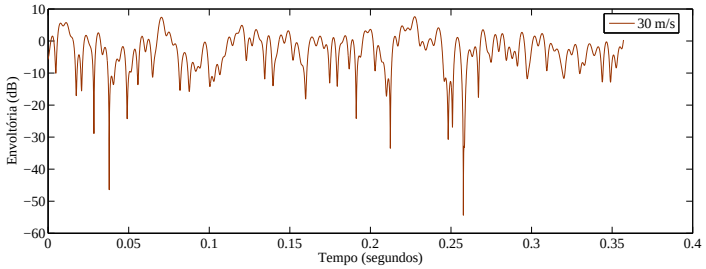
- d_s é a distância entre a antena s do conjunto de antenas da estação rádio-base e a antena de referência (para a qual $s = 1$);
- d_u é a distância entre a antena u do conjunto de antenas do móvel e a antena de referência (para a qual $u = 1$);
- $\Phi_{n,m}$ é a fase do m-ésimo *subpath* do n-ésimo multipercurso;
- $\|\mathbf{v}\|$ é o módulo do vetor velocidade;
- θ_v é o ângulo do vetor velocidade.

O modelo também apresenta funcionalidades adicionais, como, por exemplo, a modelagem dos efeitos de propagação considerando a polarização utilizada pelas antenas e a propagação em linha de visada.

Um exemplo de realização de canal para $u = s = n = 1$ é mostrado na Fig. II.2 para diferentes valores de velocidade do terminal móvel. A figura exibe o módulo da resposta $h_{u,s,n}(t)$.



(a) 5 m/s



(b) 30 m/s

Figura II.2: Exemplo de realização do canal para diferentes velocidades do móvel.