

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia eletrônica

**Simulação de beamforming em sistemas MIMO:
implementação de algoritmos adaptativos no
simulador de canais de rádio Hermespy**

Autor: Leonardo Sampaio Barros
Orientador: Prof. Dr. Daniel Costa Araújo

Brasília, DF
2023



Leonardo Sampaio Barros

**Simulação de beamforming em sistemas MIMO:
implementação de algoritmos adaptativos no simulador
de canais de rádio Hermespy**

Monografia submetida ao curso de graduação em Engenharia eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia eletrônica.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Daniel Costa Araújo

Brasília, DF

2023

Leonardo Sampaio Barros Simulação de beamforming em sistemas MIMO: implementação de algoritmos adaptativos no simulador de canais de rádio Hermespy/ Leonardo Sampaio Barros. – Brasília, DF, 2023- 70 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Daniel Costa Araújo

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB

Faculdade UnB Gama - FGA , 2023.

1. Beamforming. 2. Algoritmos Adaptativos. 3. NLMS. 4. Hermespy. 5. RLS. I.

Prof. Dr. Daniel Costa Araújo. II. Universidade de Brasília. III. Faculdade UnB

Gama. IV. Simulação de beamforming em sistemas MIMO: implementação de

algoritmos adaptativos no simulador de canais de rádio Hermespy

CDU 02:141:005.6

Simulação de beamforming em sistemas MIMO: implementação de algoritmos adaptativos no simulador de canais de rádio Hermespy

Monografia submetida ao curso de graduação em Engenharia eletrônica da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia eletrônica.

Trabalho aprovado. Brasília, DF, :

Prof. Dr. Daniel Costa Araújo
Orientador

**Prof. Dr. Daniel Mauricio Muñoz
Arboleda**
(FGA/UNB)
Convidado 1

Prof. Dr. Leonardo Aguayo
(ENE/UNB)
Convidado 2

Brasília, DF
2023

À minha família e aos meus amigos.

Agradecimentos

Agradeço primeiramente à minha família, em especial à minha mãe, Elaine, que me inspira com sua determinação e força, e que tanto batalhou pela minha educação e tornou este momento possível.

Agradeço a minha namorada, Maria Eduarda, por me auxiliar no desenvolvimento deste projeto, por sempre me acolher e apoiar com seu amor e carinho em momentos de grande dificuldade.

Agradeço aos amigos, que durante momentos de tristeza, estresse e ansiedade, me apoiaram para que eu seguisse em frente.

Agradeço ao meu orientador, professor Daniel Araújo, pela excelente orientação, e pela imensa sabedoria e ensinamentos transmitidos ao longo deste período, sempre com ampla paciência e compreensão.

Por último, mas não menos importante, agradeço aos meus companheiros de equipe da Huawei, em especial ao meu mentor, Gualter, pela generosidade com o conhecimento e pelos ensinamentos transmitidos ao longo do meu período de estágio.

A vocês e a todos os outros que me apoiaram, durante a vida e durante a conclusão deste curso, meu mais sincero obrigado!

“A tarefa não é tanto ver aquilo que ninguém viu,
mas pensar o que ninguém ainda pensou
sobre aquilo que todo mundo vê.” (Arthur Schopenhauer)

Resumo

A taxa de transmissão de dados tem crescido de maneira exponencial ao longo dos anos e, com a chegada da 5G, esse crescimento tende a ser ainda maior. Por consequência, diversas técnicas foram desenvolvidas com o intuito de melhorar os principais aspectos das comunicações sem fio, como a multiplexação espacial, feita por meio de um sistema conhecido como *Multiple Input Multiple Output* (MIMO), através da qual múltiplas antenas são utilizadas para transmissão e recepção. Uma das técnicas possíveis no cenário de MIMO é a de *beamforming*, que consiste na distribuição da energia de transmissão em uma direção de interesse, a fim de aumentar eficiência espectral sem, necessariamente, aumentar a potência do sinal transmitido, bem como, utilizar uma menor porção da banda do espectro disponível.

O uso de simuladores de canais de rádio para estudar e desenvolver novas tecnologias para redes de telecomunicações também tem sido cada vez mais comuns, e, geralmente, para realização desta técnica em específico, utilizam-se de algoritmos adaptativos. Desta forma, este trabalho busca implementar os algoritmos NLMS e RLS, desenvolvidos em python, no simulador de canais *Hermespy*, um simulador *open source* e comparar sua performance com o algoritmo MRT previamente implementado. Por fim, este trabalho comprovou a implementação dos algoritmos desenvolvidos para o simulador e validou seu funcionamento por meio dos resultados simulacionais obtidos.

Abstract

The data transmission rate has grown exponentially over the years and, with the arrival of 5G, this growth tends to be even greater. Consequently, several techniques have been developed with the aim of improving the main aspects of wireless communications, such as spatial multiplexing, carried out through a system known as Multiple Input Multiple Output (MIMO), through which multiple antennas are used for transmission. and reception. One of the possible techniques in the MIMO scenario is beamforming, which consists of distributing the transmission energy in a direction of interest, in order to increase spectral efficiency without necessarily increasing the power of the transmitted signal, as well as using a smaller portion of the available spectrum band.

The use of radio channel simulators to study and develop new technologies for telecommunications networks has also been increasingly common, and, generally, to carry out this specific technique, adaptive algorithms are used. Thus, this work seeks to implement the NLMS and RLS algorithms, developed in python, in the Hermespy channel simulator, an open source simulator, and compare its performance with the previously implemented MRT algorithm. Finally, this work confirmed the implementation of the algorithms developed for the simulator and validated its operation through the obtained simulation results.

Lista de ilustrações

Figura 1 – Tráfego de dados móveis da América Latina (Exabytes por mês), extraído de (ERICSSON, 2021)	19
Figura 2 – Canal do sistema MIMO, adaptado de (VISWANATHAN, 2014)	22
Figura 3 – Alocação de símbolos piloto no tempo, frequência e espaço, extraído de (SANTOS, 2016)	25
Figura 4 – Diagrama de blocos do <i>beamforming</i> adaptativo, adaptado de (MEENA; KABILAN, 2009)	26
Figura 5 – Cenário de simulação, adaptado de (BJÖRNSSON; HOYDIS; SANGUINETTI, 2017)	27
Figura 6 – Combinador linear, retirado de (NETO; GOMES, 2010)	33
Figura 7 – Diagrama de blocos LMS	36
Figura 8 – Diagrama de blocos RLS	38
Figura 9 – Diagrama de blocos do algoritmo proposto	40
Figura 10 – Diagrama de classes do algoritmo proposto	41
Figura 11 – Diagrama de blocos do algoritmo proposto	44
Figura 12 – Resultado para variação na quantidade de amostras de símbolos pilotos para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	46
Figura 13 – Resultado para variação na quantidade de amostras de símbolos pilotos para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	47
Figura 14 – Resultado para variação no valor de SNR para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	48
Figura 15 – Resultado para variação no valor de SNR para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	48
Figura 16 – Resultado para variação no valor de k para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	49
Figura 17 – Resultado para variação no valor de k para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	50
Figura 18 – Resultado para variação no número de antenas para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	51

Figura 19 – Resultado para variação no número de antenas para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe	51
Figura 20 – Resultado para a CDF do NLMS	53
Figura 21 – Resultado para CDF do RLS	53
Figura 22 – Resultado para CDF do MRT	53
Figura 23 – Resultado para variação na potência de treinamento para o NLMS, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe	55
Figura 24 – Resultado para variação na potência de treinamento para o RLS, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe	55
Figura 25 – Resultado para variação na potência de treinamento para o MRT, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe	56
Figura 26 – Resultado para variação no número de amostras de treinamento RLS, onde: (a) apresenta o resultado para o valor de 2 amostras de treino;(b) apresenta o resultado para o valor de 8 amostras de treino	56
Figura 27 – Resultado para variação no número de amostras de treinamento NLMS, onde: (a) apresenta o resultado para o valor de 2 amostras de treino;(b) apresenta o resultado para o valor de 8 amostras de treino	57
Figura 28 – Resultado para variação no número de amostras de treinamento MRT, onde: (a) apresenta o resultado para o valor de 2 amostras de treino;(b) apresenta o resultado para o valor de 8 amostras de treino	57
Figura 29 – Resultado para variação no valor de x e y max para o RLS, onde: (a) Apresenta o resultado para x e y igual a 10;(b) Apresenta o resultado para x e y igual a 20	58
Figura 30 – Resultado para variação no valor de x e y max para o NLMS, onde: (a) Apresenta o resultado para x e y igual a 10;(b) Apresenta o resultado para x e y igual a 20	58
Figura 31 – Resultado para variação no valor de x e y max para o MRT, onde: (a) Apresenta o resultado para x e y igual a 10;(b) Apresenta o resultado para x e y igual a 20	59
Figura 32 – Resultado para 2 usuários para o RLS	60
Figura 33 – Resultado para 3 usuários para o RLS	60
Figura 34 – Resultado para 2 usuários para o NLMS	61
Figura 35 – Resultado para 3 usuários para o NLMS	62
Figura 36 – Resultado para 2 usuários para o MRT	63
Figura 37 – Resultado para 3 usuários para o MRT	64

Lista de tabelas

Tabela 1 – Numerologias comuns para a rede 5G	24
Tabela 2 – Parâmetros fixos para a simulação simplificada	45
Tabela 3 – Parâmetros Variados para a simulação simplificada	46
Tabela 4 – Valores de meio percentil de capacidade e eficiência espectral para mil iterações	54
Tabela 5 – Parâmetros Variados para a simulação no <i>Hermespy</i>	54
Tabela 6 – Parâmetros fixos para a simulação no <i>Hermespy</i>	54
Tabela 7 – Valores de capacidade e eficiência espectral RLS para múltiplos usuários	61
Tabela 8 – Valores de capacidade e eficiência espectral NLMS para múltiplos usuários	62
Tabela 9 – Valores de capacidade e eficiência espectral MRT para múltiplos usuários	63

Lista de abreviaturas e siglas

3GPP	<i>Third Generation Partnership Project</i>
AWGN	<i>Additive White Gaussian Noise</i>
CDF	<i>Cumulative Distribution Function</i>
CP	<i>Cyclic Prefix</i>
CSI	<i>Channel State Information</i>
DOA	<i>Direction of Arrival</i>
ERB	Estação Rádio Base
FIR	<i>Finite Impulse Response</i>
Hermespy	<i>Heterogeneous Mobile Radio Simulator python</i>
IIR	<i>Infinite Impulse Response</i>
ISI	<i>Intersymbol Interference</i>
LMS	<i>Least Mean Squares</i>
LoS	<i>Line of Sight</i>
MIMO	<i>Multiple Input Multiple Output</i>
MISO	<i>Multiple Input Single Output</i>
MRT	<i>Maximum Ratio Transmission</i>
MSE	<i>Mean Square Error</i>
NLMS	<i>Normalized Least Mean Squares</i>
OFDM	<i>Orthogonal Frequency Division Multiplexing</i>
OFDMA	<i>Orthogonal Frequency Division Multiple Access</i>
PRB	<i>Physical Resource Block</i>
RLS	<i>Recursive Least Squares</i>
SNR	<i>Signal to Noise Ratio</i>

ULA	<i>Uniform Linear Array</i>
UE	<i>User Equipment</i>
URA	<i>Uniform Rectangular Array</i>
ITU	<i>International Telecommunications Union</i>

Lista de símbolos

θ	Ângulo de visada
$\Delta\theta$	Variação no ângulo de visada
λ	Comprimento de onda
μ_{lms}	Passo de convergência
μ_{rls}	Fator de esquecimento
K	Fator de Rice
\mathbf{H}	Matriz do Canal
\mathbf{w}	Vetor de coeficientes de ajuste
\mathbf{W}	Matriz de coeficientes de ajuste
\mathbf{x}	Vetor do sinal enviado
\mathbf{X}	Matriz do sinal enviado
C	Eficiência espectral
Rb	Capacidade do sistema
\mathbf{p}	Vetor de correlação cruzada
\mathbf{R}	Matriz de auto-correlação
\mathbf{V}_{los}	Sinal recebido em linha de visada
\mathbf{V}_{nlos}	Sinal recebido dos multipercursos
e	Erro estimado
d	Sinal de interesse
y	Sinal obtido

Sumário

1	INTRODUÇÃO	17
1.1	Justificativa	18
1.2	Objetivo geral	19
1.3	Objetivos específicos	20
1.4	Organização do documento	20
2	SISTEMAS MIMO MASSIVO	21
2.1	Descrição geral	21
2.2	Características de redes 5G	23
2.3	Técnica de conformação de feixes	25
2.4	Cenários	26
2.4.1	Cenário Simplificado	27
2.4.2	Cenário do simulador	29
2.4.3	Valores de interesse	30
3	ALGORITMOS ADAPTATIVOS	31
3.1	Introdução	31
3.2	Filtro de Wiener	32
3.3	Algoritmos Adaptativos	32
3.3.1	Introdução	32
3.3.2	Algoritmo LMS	33
3.3.3	Algoritmo NLMS	35
3.4	Algoritmo RLS	36
3.5	Algoritmo MRT	38
3.6	Algoritmo proposto	39
3.6.1	Algoritmo simplificado	39
3.6.2	Algoritmo <i>Hermespy</i>	40
3.6.2.1	Ambiente principal	41
3.6.2.2	Classe <i>simulation</i>	42
3.6.2.3	Classe <i>beamformingbaseclass</i>	43
4	SIMULAÇÕES E RESULTADOS	45
4.1	Simulações simplificadas	45
4.1.1	Variação do número de símbolos pilotos	46
4.1.2	Variação na relação sinal-ruído	47
4.1.3	Variação no fator de Rice	49

4.1.4	Variação no número de antenas	50
4.2	Simulações no <i>Hermespy</i>	52
4.2.1	Variação na potência do sinal	55
4.2.2	Variação no número de PRBs de treinamento	56
4.2.3	Variação nos desvios em x e y	57
4.2.4	Variação no número de usuários	59
4.3	Conclusões do capítulo	64
5	CONCLUSÃO	66
5.1	Futuros trabalhos	66
	REFERÊNCIAS	68

1 Introdução

No dia 6 de julho de 2022, um marco ocorreu na história das comunicações móveis no Brasil: a rede 5G foi ativada em Brasília. A quinta geração de redes móveis e de banda larga (5G) promete revolucionar não só o mercado de telecomunicações, mas também, o mercado de tecnologia como um todo, haja vista o 5G propor uma conexão de alta confiabilidade, alta conectividade e baixa latência, além das altíssimas taxas de transmissão de dados (AGIWAL; ROY; SAXENA, 2016).

Conforme estabelecido pelo *3rd Generation Partnership Program* (3GPP), órgão internacional responsável por definir parâmetros para redes de telecomunicações, as metas para o 5G são: latência de até 0,5 ms, conectividade de até 1 milhão de aparelhos por km², ultra confiabilidade e conexão de 1 até 10 Gbps (3GPP, 2018a).

Nesse viés, é vital que essas metas sejam atingidas para que o 5G possa, de fato, proporcionar a revolução tecnológica esperada, visto que, para aplicações críticas como carros autônomos, cirurgias teleguiadas, biossensores, entre outras, a menor das alterações pode acarretar em prejuízos financeiros enormes e até mesmo na perda de vidas humanas.

A partir disso, novas técnicas foram desenvolvidas a fim de melhorar diversos aspectos do canal, como a relação sinal/ruído, a melhora da eficiência espectral, a interferência multipercurso, a capacidade de usuários conectados ao mesmo tempo em uma célula, entre outros. Uma dessas técnicas consiste no emprego de múltiplas antenas, no sistema conhecido como MIMO, do inglês *Multiple Input Multiple Output*, que também possui sistemas considerados como MIMO massivo devido ao amplo número de elementos irradiantes em apenas uma estação de rádio base (MARZETTA, 2015).

Há também o emprego de outras estratégias que buscam resolver os problemas supracitados. Entre elas, destacam-se a densificação de células, conhecida como *small cells*, e a exploração de frequências de ondas milimétricas, as *mm-Waves*, que são subutilizadas ou não utilizadas em redes de comunicação tradicionais. Dentro de cada uma dessas áreas, há diversas ramificações que buscam aprimorar as características dessas técnicas. Em cenários de transmissão com múltiplas antenas transmissoras, utiliza-se a técnica de *beamforming*, ou conformação de feixe em português, para ampliar o ganho em diversidade espacial. Essa técnica é utilizada para direcionar a energia de transmissão em uma direção específica, melhorando a eficiência espectral sem aumentar a potência do sinal transmitido ou utilizar mais largura de banda.

A técnica de *beamforming* pode ser realizada utilizando sistemas de defasadores e receptores analógicos, híbridos ou digitais. Dentre esses sistemas, os digitais apresentam a vantagem de permitir a implementação de processos adaptativos por meio de algoritmos

que visam minimizar o erro quadrático médio entre o sinal obtido e o sinal de interesse. Embora a solução híbrida também possua essa característica e necessite de menor custo financeiro, energético e computacional, ela pode aumentar a complexidade do sistema (SOHRABI; YU, 2016). Os sistemas analógicos apresentam limitações de flexibilidade e precisão, além de serem menos eficientes em termos de energia e largura de banda. Os sistemas digitais, por outro lado, permitem a implementação de algoritmos adaptativos, como o RLS e o NLMS, para realizar o *beamforming* de forma mais eficiente e precisa.

Devido às novas tecnologias introduzidas com o 5G, a avaliação performática e o planejamento destas redes se tornaram procedimentos bastante desafiadores e de grande demanda computacional (GKONIS; PANAGIOTIS; KAKLAMANI, 2020). Desta forma, nos últimos anos, dezenas de simuladores de canais de rádio foram desenvolvidos, cada um possuindo pontos fracos e fortes, a partir das escolhas de arquitetura de *software* e de modelos e algoritmos de comunicação implementados. Em comparação com os demais simuladores existentes, o *Heterogeneous Radio Mobile Simulator In Python (Hermespy)*, desenvolvido pelo instituto *Barkhausen*, apresenta como vantagem uma vasta opção de recursos de simulação já implementados, além de ser gratuito e *open source* (ADLER; KRONAUER; BARRETO, 2022).

Dessa forma, este trabalho busca, por meio da implementação dos blocos de algoritmos adaptativos de *Least Mean Squares* (LMS) e *Recursive Least Squares* (RLS), contribuir com o aprimoramento do simulador de canais de rádio *Hermespy*. Serão realizadas simulações para validar seu funcionamento e avaliar o desempenho dos mesmos quando comparados ao algoritmo *Maximum Ratio Transmitter* (MRT), por se tratar de uma funcionalidade preexistente no simulador. O cenário escolhido como base para realização das simulações é o de transmissão *uplink*, ou seja, a informação é enviada do terminal do usuário para a estação rádio base (ERB), em campo distante e na linha de visada, constituindo um cenário onde não há obstáculos interceptando a trajetória do sinal entre a ERB e o usuário. As características de transmissão serão equivalentes às estabelecidas por órgãos reguladores para redes 5G.

1.1 Justificativa

Na década que compreende os anos de 2011 a 2021, a taxa de tráfego de dados móveis cresceu cerca de 300 vezes e a previsão é de que, até o ano de 2027, as conexões deverão crescer cerca de três vezes mais, com até metade delas sendo realizadas através da rede 5G (ERICSSON, 2021).

Nos últimos 6 anos, a demanda cresceu por volta de 42% ao ano, fazendo-se necessário o uso de novas técnicas que permitam melhorar a eficiência na transmissão de dados e que sejam de simples implementação. Assim, foram desenvolvidas tecnologias como o

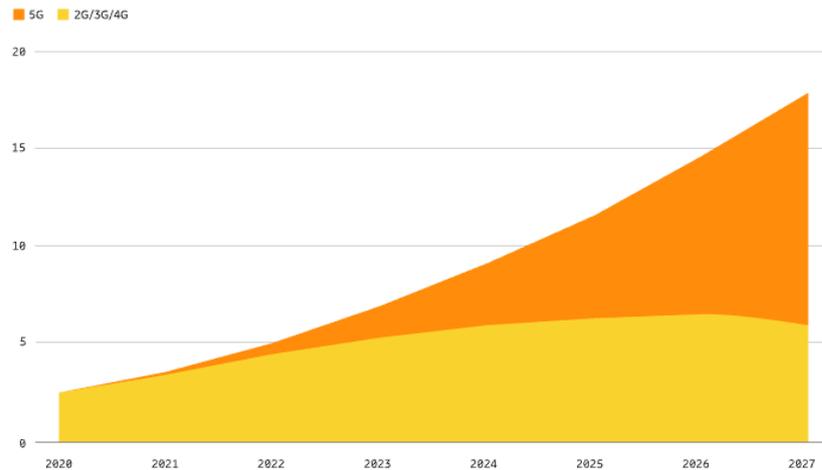


Figura 1 – Tráfego de dados móveis da América Latina (Exabytes por mês), extraído de (ERICSSON, 2021)

MIMO massivo, que apresenta como vantagem em relação ao sistema tradicional *Single Input Single Output* (SISO), o melhor uso da eficiência espectral, o ganho de diversidade espacial, a melhor eficiência energética, entre outros benefícios. Além disso, em conjunto com a implementação do MIMO, podem ser utilizadas técnicas de processamento espacial para potencializar, ainda mais, o desempenho destes sistemas.

No quesito de multiplexação espacial, ou seja, da divisão dos usuários também no domínio do espaço, o uso da técnica de *beamforming* permite maximizar a transmissão através do direcionamento de sinais de rádio para uma posição específica. O feixe direcional pode ser ajustado de maneira dinâmica ao se utilizar algoritmos adaptativos, possibilitando, assim, ajuste em tempo real deste, redução do tempo de convergência do processo, entre outros.

Logo, o objetivo deste projeto é contribuir com o aprimoramento do simulador de canais de rádio *Hermespy* e avaliar a performance de algoritmos adaptativos no processo de conformação de feixes. Isso será alcançado através do desenvolvimento de novos recursos de simulação para *Hermespy*, um simulador de canais de rádio *opensource*, ou seja, com código fonte aberto, desenvolvido em *python*. Com isso, espera-se contribuir para a evolução do simulador e fornecer uma melhor compreensão do funcionamento dos algoritmos adaptativos na conformação de feixe, ou *beamforming*.

1.2 Objetivo geral

Este trabalho busca implementar novas funcionalidades de técnicas de *beamforming* adaptativo para o simulador *Hermespy*, e avaliar suas performances comparando-as com uma funcionalidade preexistente em um cenário de enlace MIMO com transmissão *uplink* em campo distante e na linha de visada, utilizando os algoritmos NLMS e RLS. Para isso,

estes foram desenvolvidos em python e inseridos no simulador de canais de comunicações móveis, para com o aprimoramento do mesmo e, ao mesmo tempo, verificar o desempenho dos algoritmos.

1.3 Objetivos específicos

Os objetivos específicos referentes a este trabalho são:

- Contextualizar os sistemas classificados como MIMO e as características de transmissão de redes 5G.
- Apresentar a técnica de *beamforming*.
- Descrever o funcionamento dos algoritmos adaptativos NLMS e RLS e implementá-los no código do *Hermespy*.
- Realizar simulações de *beamforming* em cenário de transmissão *uplink* em linha de visada, variando os parâmetros de número de usuários e de valores característicos da transmissão de redes 5G.
- Avaliar os impactos dessa variações e o desempenho dos algoritmos no processo de conformação de feixe, comparando-os com os resultados obtidos pela técnica de MRT.

1.4 Organização do documento

Além do Capítulo 1 da Introdução, o restante deste trabalho será assim estruturado:

- Capítulo 2 – Abordagem dos aspectos que compõem um cenário caracterizado como MIMO e suas vantagens em relação ao sistema tradicional, as características inerentes às redes 5G e o conceito de *beamforming*.
- Capítulo 3 – Discussão das técnicas de filtragem adaptativa e sua aplicação no processo de conformação de feixes e descrição dos algoritmos utilizados neste trabalho.
- Capítulo 4 - Apresentação dos resultados obtidos a partir das simulações, com o objetivo de analisar o desempenho das técnicas empregadas e os efeitos provocados pelas alterações de determinados parâmetros, bem como validar a implementação dos algoritmos no simulador.
- Capítulo 5 - Exposição das conclusões realizadas a partir da análise deste trabalho.

2 Sistemas MIMO massivo

Este capítulo é destinado ao estudo do funcionamento de sistemas com múltiplas antenas e das características de redes 5G, visto tal entendimento ser fundamental à compreensão do processo de conformação de feixe a ser abordado, futuramente. Também busca-se descrever as possíveis implicações dos cenários utilizados para as simulações.

2.1 Descrição geral

Há muito tempo, as redes de telecomunicações enfrentam o desafio de acompanhar o progressivo aumento das taxas de transmissão de dados, enquanto a disponibilidade de espectro eletromagnético para utilização permanece sempre o mesmo (MARZETTA, 2015).

Nesse viés, algumas técnicas foram desenvolvidas a fim de combater este problema, como as *mm-Wave*, que buscam utilizar as frequências ainda não utilizadas, ou subutilizadas do espectro eletromagnético, as *Small cells*, que tem por objetivo atender uma porção de região menor, e os sistemas com múltiplos pontos de acesso, conhecido como MIMO.

O sistema MIMO consiste na utilização de um arranjo de múltiplas antenas no sistema de comunicação na transmissão e na recepção que, geralmente, operam de acordo com o protocolo síncrono TDD (BJÖRNSON; HOYDIS; SANGUINETTI, 2017).

O emprego do MIMO nos cenários de telecomunicações proporciona ganho em diversidade espacial em performance, sem necessariamente aumentar a potência irradiada e sem utilizar mais recursos espectrais. O ganho em diversidade espacial reflete na redução da distorção do sinal proporcionado pelo caráter multipercorso do enlace de comunicação (SANTOS, 2016). Isto ocorre porque as antenas são idealmente espaçadas entre si por uma distância de decorrelação, fazendo com que o sinal e suas múltiplas réplicas alcancem o destino por diversos percursos independentes, proporcionando uma maior probabilidade de uma dessas vertentes do sinal chegar com menor presença de erro. Já a não necessidade de aumentar a potência de transmissão ou de utilizar um maior recurso eletromagnético resulta na maior eficiência espectral e na redução do gasto energético.

A multiplexação espacial com múltiplas antenas oferece uma série de vantagens além das mencionadas anteriormente. Uma delas é a formação de pequenos feixes de transmissão independentes, que ocupam uma porção menor da banda do canal. Isso significa que o uso de múltiplas antenas permite uma utilização mais eficiente da largura de banda disponível.

Além disso, cada antena recebe uma réplica independente do sinal, o que permite

uma separação dos usuários não apenas no domínio tempo-frequência, mas também no espaço. Isso significa que os feixes formados podem ser direcionados a cada usuário, de acordo com sua posição, aumentando a eficiência da transmissão de dados.

Em resumo, a multiplexação espacial com múltiplas antenas permite a utilização mais eficiente da largura de banda disponível e a separação mais precisa dos usuários, resultando em uma melhor qualidade de serviço para cada usuário individual.

Há duas possibilidades para os sistemas MIMO, o chamado MIMO de ponto a ponto e o MIMO de múltiplos usuários (*Multiple Users MIMO*, MU-MIMO). O primeiro, a proposta mais simples, consiste em múltiplas antenas, tanto na estação de rádio base (ERB), quanto no terminal do usuário. O segundo consiste em uma ERB com múltiplas antenas que transmitem e recebem dados de vários equipamentos do usuário, estes com apenas uma antena, que em conjunto representariam as antenas do MIMO ponto a ponto. Assim, em ambos, é formado um canal \mathbf{H} composto de uma matriz de \mathbf{m} antenas transmissoras por \mathbf{n} antenas receptoras. Dessa forma, tem-se um modelo de canal visto na Figura 2.

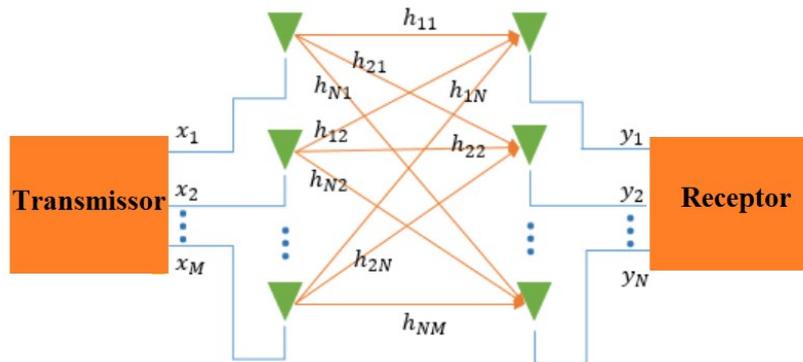


Figura 2 – Canal do sistema MIMO, adaptado de (VISWANATHAN, 2014)

Então, é possível descrever o sinal recebido \mathbf{Y} através da matriz de transmissão \mathbf{H} , do vetor de entrada \mathbf{x} e do vetor de ruído AWGN \mathbf{n} , pela Equação 2.1

$$\mathbf{Y} = \mathbf{H}\mathbf{x} + \mathbf{n}. \quad (2.1)$$

Em ambos os casos, como a transmissão de dados é feita ao mesmo tempo e na mesma frequência para os diferentes usuários, é necessário tanto à ERB, quanto ao UE, ter, previamente, as informações do estado do canal, chamadas de CSI (do inglês *Channel State Information*), para que a utilização do MIMO seja efetiva. A aquisição dessas informações é feita através do envio de símbolos pilotos diferentes para cada usuário e que também precisam ser conhecidos pela base receptora. Desse modo, quando o sinal é processado, a informação é extraída para cada usuário, separadamente.

Há ainda o cenário de MIMO massivo, no qual utiliza-se um maior número de antenas em relação ao número de usuários (MARZETTA, 2010), o que proporciona um ganho ainda maior, chegando a um incremento na capacidade de até dez vezes, e no ganho de eficiência energética relativa à potência irradiada da ordem de 100 vezes, ambos resultado da multiplexação espacial em larga escala (LARSSON et al., 2014).

2.2 Características de redes 5G

Cada geração de tecnologia de redes móveis apresenta características de canal que são ora herdadas de gerações anteriores, ora acrescentadas para promover avanços na qualidade e capacidade de transmissão de dados.

O *beamforming* começou a ser utilizado com ERBs, ainda na quarta geração, possuindo, em sua maioria, sistemas de 4 antenas. Porém, seu verdadeiro potencial deve ser alcançado da quinta geração em diante, já que esta tem como uma de suas bases a utilização do MIMO massivo, como dito anteriormente, com ERBs possuindo arranjos de 64 ou mais antenas. Assim, este estudo adotará como característica padrão as inerentes aos canais da quinta geração.

O 3GPP e o *International Telecommunications Union* (ITU) são os principais órgãos internacionais que estabelecem regras e padrões para as características dos enlaces de comunicação de cada geração, com vistas aos valores-chaves a serem utilizados em cada tecnologia. Valores como as bandas de frequência utilizadas, os espaçamentos das subportadoras, o tamanho dos *time slots* de cada recurso, entre outros, são estabelecidos globalmente para facilitar o desenvolvimento e a implementação de novas tecnologias.

Nas redes móveis, desde a quarta geração, a informação é transmitida através da técnica de Multiplexação por divisão de frequências ortogonais (OFDM, do inglês, *Orthogonal Frequency Division Multiplexing*). Como princípio básico, a OFDM permite que os dados sejam separados em várias sequências de menor taxa, transmitidas por subcanais paralelos, ao invés da maneira tradicional, onde os dados eram enviados em um único fluxo, chamado de *stream*. Assim, no OFDM, cada subcanal possui uma subportadora, de maneira que a vazão de dados do sistema com várias subportadoras seja a mesma que a do sistema com portadora única. As subportadoras estão espaçadas entre si a uma certa largura de banda que, para o 4G, foi estabelecida em 15 kHz. Porém, a ortogonalidade dos subcanais permite a sobreposição espectral das subportadoras, viabilizando o maior fluxo de dados, como mencionado, e apresentando grandes vantagens para o sinal, como menor sensibilidade a ruídos, a interferência inter-simbólica (ISI), e a multiplicidade de caminhos devido ao maior tempo de duração de cada símbolo (MENDES, 2007).

Cada símbolo OFDM é precedido por um prefixo cíclico (CP), que é a cópia de uma parcela do final do símbolo modulado. Este recurso é utilizado para reduzir ainda

mais a ISI, e pode ser modulado como um CP normal ou estendido.

Para transmissão nas redes 4G e 5G, a OFDM foi utilizada junto a uma técnica que funciona como uma extensão da original, o acesso múltiplo por divisão ortogonal de frequência (OFDMA). Nesta, ao contrário da OFDM, onde cada usuário utilizava um subcanal, os usuários são alocados de maneira dinâmica nos subcanais, permitindo assim, que eles transmitam e recebam sinais ao mesmo tempo (MENDES, 2007).

A transmissão no domínio do tempo é feita no formato de quadros. Um quadro, na rede 4G, tem duração de 10 ms, sendo dividido em 10 sub-quadros com duração individual de 1 ms, possuindo, no mínimo, dois *slots* de 0,5 ms.

A principal diferença da rede 5G em comparação com a antiga geração é a variedade de espaçamentos entre as sub-portadoras, que influenciam na duração de cada um desses *slots*. Desta forma, temos que um aumento no espaçamento entre as subportadoras ocasiona uma diminuição na duração de cada *slot*, permitindo assim que uma maior quantidade destes recursos exista em cada sub-quadro. Cada *slot* possui uma quantidade fixa de 7 símbolos OFDM, para o CP normal, e 6 símbolos para o CP estendido. Daí resulta a formação de um bloco de recurso (RB), que é a menor quantidade de recursos possível de ser alocada para um usuário, e representa a duração 12 subportadores consecutivas na frequência pela duração de 1 *slot* no tempo.

A tabela 1 apresenta o número de *slots*, de símbolos no prefixo cíclico normal e de subportadoras em um RB de acordo com cada numerologia estabelecida pelo 3GPP para o 5G.

μ	$\Delta f = 2^\mu \cdot 15$ [kHz]	$N_{slot}^{símbolos}$	$N_{slot}^{frames\mu}$	$N_{slot}^{subframes\mu}$
0	15	14	10	1
1	30	14	20	2
2	60	14	40	4
3	120	14	80	8
4	240	14	160	16
5	480	14	320	32
6	960	14	640	64

Tabela 1 – Numerologias comuns para a rede 5G

Como mencionado anteriormente, uma das grandes vantagens da utilização dos sistemas MIMO é a possibilidade de alocação dos recursos também no espaço. Para os sistemas SISO nas redes de quarta geração, a transmissão era feita de maneira dinâmica através da técnica de agendamento, na qual a cada sub-quadro o *scheduler* controlava quais dispositivos poderiam transmitir ou receber sinais e em qual recurso de frequência (SKONDRAS et al., 2015). Caso os usuários realizassem transmissões o mesmo tempo e na

mesma frequência, os sinais interfeririam uns nos outros. Porém, para os sistemas MIMO, cada antena receberá uma réplica deste sinal. Desta forma, além de separar os usuários por recurso de frequência e por tempo, os usuários são ,também, separados espacialmente nos arranjos de antenas de acordo com o índice da antena transmissora ou receptora, não sendo necessária a realização do agendamento.

Um exemplo de como os símbolos pilotos podem ser alocados no tempo, frequência e espaço pode ser visto em 3.

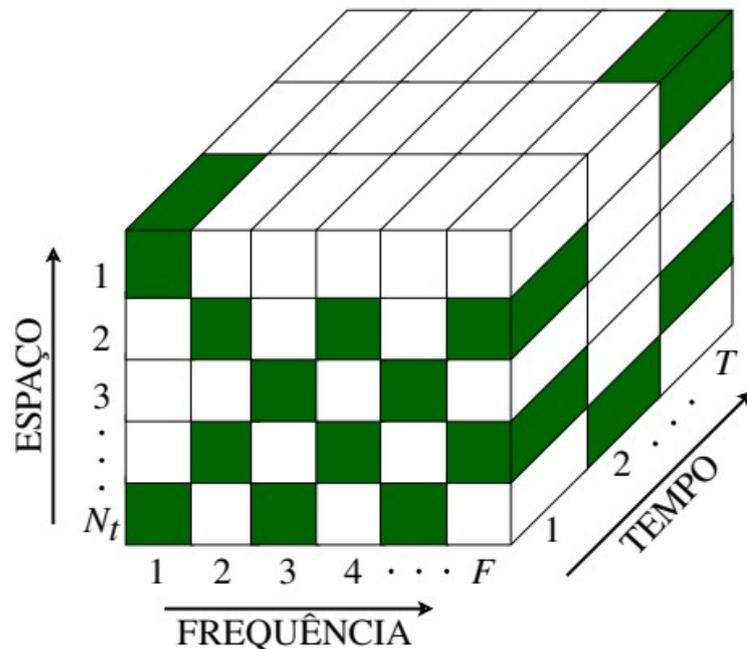


Figura 3 – Alocação de símbolos piloto no tempo, frequência e espaço, extraído de (SANTOS, 2016)

Logo, por padrão, observa-se a quantidade de símbolos OFDM, os valores padrões de espaçamento entre as sub-portadoras e a quantidade de PRBs presentes em cada *frame* de transmissão. Esses valores servirão de base para realização das simulações que serão realizadas adiante.

2.3 Técnica de conformação de feixes

Conforme abordado anteriormente, nos sistemas MIMO, a transmissão ocorre formando pequenos feixes, que são direcionados ao usuário e garantem menor ocupação da banda do canal e maior taxa de transmissão de dados. Desta maneira, a técnica de conformação de feixe, ou em inglês *beamforming*, é uma técnica de multiplexação espacial que, por meio do processamento de sinais, realiza o direcionamento da energia de transmissão desses feixes na direção desejada (VEEN; BUCKLEY, 1988).

Para realizar o direcionamento do feixe nos sistemas MIMO, são utilizados pesos fixos e adaptativos. Os fixos são utilizados como constantes que afetam cada elemento irradiante. Já os adaptativos, tema de estudo deste trabalho, constituem-se de algoritmos recursivos capazes de realizar um autoajuste em tempo real, para assim alterar o padrão de irradiação. Tais algoritmos serão abordados no próximo capítulo.

Considerando um arranjo de antenas inteligentes, que possuem um mecanismo identificador da a direção de chegada dos sinais (*Direction of arrival*, DOA), e a utilização de um algoritmo adaptativo para ajuste dos pesos, o sistema será capaz de direcionar o feixe emitido no ângulo da DOA. Assim, o diagrama de como seria o sistema de conformação de feixe adaptativo pode ser visto na Figura 4.

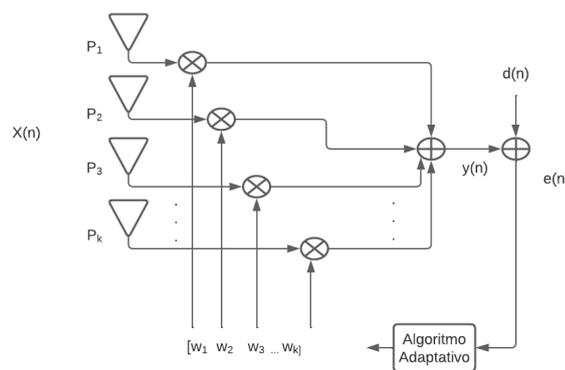


Figura 4 – Diagrama de blocos do *beamforming* adaptativo, adaptado de (MEENA; KABILAN, 2009)

Os pesos definidos pelo algoritmo são responsáveis por gerar a alteração de fase do sinal e também da amplitude, o que aponta o sinal irradiado para a direção desejada. Este processo pode ser realizado tanto na transmissão do sinal, quanto em sua recepção.

O *beamforming* adaptativo apresenta ainda uma grande vantagem em comparação a outras técnicas de conformação de feixe, dado a ele não necessitar da estimação da frequência do canal para realizar a conformação do feixe. À medida em que as amostras de símbolos pilotos chegam, ele automaticamente ajusta o direcionamento do feixe através da estimação do canal, o que proporciona maior rapidez no processo.

2.4 Cenários

Foram utilizados dois cenários diferentes para desenvolvimento e validação dos algoritmos. O primeiro consiste em um cenário simples que será utilizado de base para construção inicial dos algoritmos e verificação de seu funcionamento na formação do feixe. O segundo é um cenário fornecido pela biblioteca de canais do simulador *Hermespy*, que simula um canal mais realista, e para o qual os algoritmos serão implementados como funcionalidade e terão sua eficácia novamente verificada através dos parâmetros

de eficiência espectral, capacidade do sistema, capacidade do sistema por usuário e o resultado do feixe.

2.4.1 Cenário Simplificado

Neste projeto, será utilizado um sistema MU-MIMO em um cenário de transmissão *uplink* em campo distante, com o receptor formado por um arranjo planar uniforme de antenas. Porém, para introdução dos conceitos, será utilizado um arranjo linear uniforme (*Uniform Linear Array*, ULA) de antenas, uma vez que os cálculos serão os mesmos para ambos os eixos do arranjo. As implicações deste cenário são: as ondas eletromagnéticas se comportam como ondas planas em linha de visada; o UE contém apenas uma antena; o protocolo usado é o TDD. Uma figura de como se caracteriza esse cenário pode ser vista em 5.

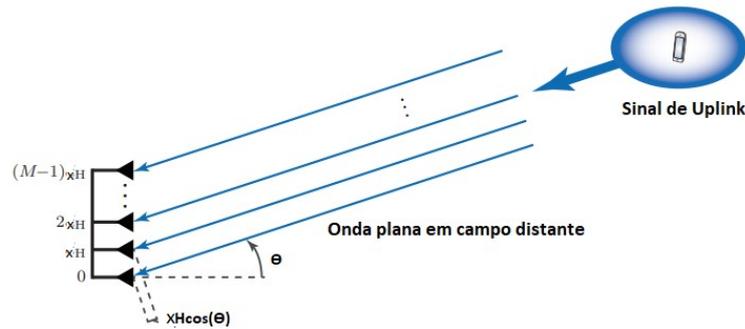


Figura 5 – Cenário de simulação, adaptado de (BJÖRNSON; HOYDIS; SANGUINETTI, 2017)

O sinal recebido pelas m antenas é vetorial e constituído por algumas variáveis. Inicialmente, para a aquisição das CSI por parte da ERB, uma sequência de tamanho n de símbolos piloto é enviada para que essas informações sejam adquiridas. Em seguida, tem-se o vetor do sinal emitido. Dado o cenário descrito acima, o vetor de transmissão pode ser dado por

$$\mathbf{v}(\theta) = [e^{-j\frac{2\pi}{\lambda}x_1 \cdot \cos(\theta)} \dots e^{-j\frac{2\pi}{\lambda}x_{(i-1)} \cdot \cos(\theta)}] \quad (2.2)$$

Neste caso, d é a distância entre as antenas de maneira que elas sejam descorrelacionadas, λ é o comprimento de onda do sinal enviado, x é a distância entre os elementos irradiantes dispostos linearmente, i é o índice que representa a posição do elemento no arranjo, e θ é o ângulo da linha de visada.

Deve-se considerar, também, o efeito multipercurso do enlace de comunicações. Desta forma, sabe-se que o sinal também chega por outros caminhos além do direto, constituindo l componentes, que estão deslocadas por um ângulo $\Delta\theta$ caracterizando, o **vnlos**:

$$\mathbf{vnlos}(\theta + \Delta\theta) = [e^{-j\frac{2\pi}{\lambda}xi \cdot \cos(\theta + \Delta\theta)} \dots e^{-j\frac{2\pi}{\lambda}x(i-1) \cdot \cos(\theta + \Delta\theta)}] \quad (2.3)$$

Apesar deste caráter multipercurso, a componente em linha direta é superior às demais do canal, podendo ser melhor exemplificado pelo modelo de distribuição de Rice, pois, nele, a componente em *LoS* sobrepõe o sinal das demais (RODRIGUES, 2003). Existe um fator, chamado de fator de Rice (fator Riciano), capaz de correlacionar a potência da componente direta com a potência das demais componentes multipercurso. Este fator é dado por

$$k(dB) = 10 \log \frac{A^2}{2\sigma^2} \quad (2.4)$$

k é o fator Riciano, A é a amplitude de pico do sinal dominante e σ é a tensão do pacote de sinal recebido. Para este cenário, o valor do fator Riciano é convertido de dB para uma variável linear, através da transformação

$$K = 10^{k(dB)/10} \quad (2.5)$$

Em seguida, deve-se considerar a rotação de fase realizada pelo sinal durante seu percurso, bem como o ruído Gaussiano branco aditivo (*Additive Gaussian White Noise*, AWGN). Assim, dada a rotação como $e^{j\phi}$, a relação de sinal-ruído (SNR) igual a $10dB$, que é utilizada para constituição da matriz de ruído \mathbf{N} , e o fator de Rice como $k = 1dB$, será utilizado o resultado das Equações 2.2 e 2.3 para obter o sinal da componente principal \mathbf{X} recebido pela ERB.

$$\mathbf{X} = \sqrt{\frac{K}{K+1}} \cdot e^{j\phi} \cdot \mathbf{v}_{los} \cdot \mathbf{d}^T + \sqrt{\frac{1}{k+1}} \cdot e^{j\phi} \cdot \mathbf{v}_{nlos} \cdot \mathbf{d}^T + \mathbf{N} \quad (2.6)$$

A Equação 2.6 será utilizada como base para simulação dos sinais, e estes para realização da posterior filtragem adaptativa. Ressalta-se o fato de este ser um modelo simplificado de canal, no qual foram desconsiderados outros tipos de interferência, como a intersimbólica (ISI) e, a contaminação de símbolos pilotos, entre outros efeitos diversos presentes nos enlaces reais de comunicações móveis.

2.4.2 Cenário do simulador

O cenário escolhido para utilização do simulador *Hermespy* foi o de Macro Urbano em linha de visada. Portanto, será aplicada a funcionalidade de *cluster delay line urban macrocells*.

O Macro Urbano em linha de visada tem seus parâmetros baseados na especificação 36.873 do 3GPP (3GPP, 2018b), que predefine valores para várias características do canal, entre elas, a altura da ERB, superior a 25 m, e do UE, entre 1,5 m a 22,5 m, assumindo-se que ele está em linha de visada.

Além do canal, o simulador possui pré-definidas as características de simulação dos dispositivos, tanto os do usuário, quanto os da ERB, e ainda, as das antenas utilizadas.

Os dispositivos são formados com as informações de posição, orientação, número de antenas e frequência central da portadora. Já as antenas escolhidas foram as ideais, porém, aqui elas estão dispostas de maneira a formar um arranjo uniforme retangular (*Uniform Rectangular Array*, URA), o que faz com que elas estejam dispostas tanto no eixo x , quanto no eixo y .

Dessa maneira, o sinal recebido assume outra forma. O espaçamento entre as antenas ainda é o mesmo, tanto na horizontal, quanto na vertical, mas a DOA é representada por uma coordenada azimutal φ e elevação θ . Dessa forma, o vetor de sinais recebidos pelos elementos da antena pode ser descrito como

$$\mathbf{X} = \mathbf{h}\mathbf{s}^T + \mathbf{N} \quad (2.7)$$

Neste caso, o vetor \mathbf{h} é o vetor de apontamento, e, \mathbf{s} e \mathbf{N} continuam sendo, respectivamente, o vetor de símbolos recebidos e a matriz de ruído. Como os sinais são recebidos em fases diferentes pelas antenas, é possível estimar sua DOA. O vetor \mathbf{h} pode ser descrito por

$$h = \sum_{i=1}^T \alpha_i a(\varphi_i, \theta_i) \quad (2.8)$$

Considerando o sistema de coordenadas cartesianas, um elemento que possua as coordenadas (m,n) , pode-se descrever a fase deste como

$$\phi_{m,n}(k)(\varphi_k, \theta_k) = \frac{2\pi}{\lambda} (d_y m \cos \theta_k \sin \varphi_k + d_z n \sin \theta_k) \quad (2.9)$$

Utilizando a Equação 2.9, temos então que o vetor de apontamento pode ser dado por:

$$\alpha_{m,n}(\varphi_k, \theta_k) = e^{j\phi_{m,n}(k)(\varphi_k, \theta_k)} \quad (2.10)$$

Além das características descritas para o cenário simplificado de desvio de percurso e desvanecimento Riciano, o cenário ainda leva em consideração as características de transmissão de redes 5G descritas anteriormente, como transmissão feita através de símbolos OFDM e o espaçamento flexível entre as subportadoras.

2.4.3 Valores de interesse

Além dos resultados gráficos, é avaliada a performance dos algoritmos adaptativos no processo de conformação de feixes, através dos valores de eficiência espectral, capacidade do sistema e capacidade do sistema por usuário.

A eficiência espectral é uma medida da capacidade de um sistema de utilizar de forma eficiente a largura de banda disponível, cujo cálculo é dado pela razão entre a capacidade do sistema R_b e a taxa de amostragem do canal R_s .

Cada sinal de $SINR$ é tridimensional e composto por 12 subportadoras por n blocos de recurso e por m usuários. Então, para obter a capacidade total do sistema, basta realizar o somatório da capacidade nas três dimensões. Com o resultado da capacidade, podemos calcular a eficiência espectral através da Equação 2.11

$$C = \frac{R_b}{B} \quad (2.11)$$

Ou seja, a eficiência espectral C é dada pela razão entre a capacidade do sistema R_b pela largura de banda total B . Considerando que o canal é do tipo Gaussiano determinístico, com a predominância do AWGN, podemos considerar que, em cada dimensão, a capacidade do sistema é dada por (VERDU, 2002)

$$R_b = R_s \cdot \log_2(1 + SINR) \quad (2.12)$$

Por fim, para os casos com mais de um usuário, é necessário obter também a capacidade do sistema por usuário, dado pelo resultado do somatório bidimensional das 12 subportadoras por n PRBs.

3 Algoritmos Adaptativos

Neste capítulo, abordaremos o conceito de filtragem adaptativa para melhor entender seu funcionamento e sua aplicação na técnica de conformação de feixes, bem como serão descritos os códigos utilizados como base para realização das simulações deste projeto.

3.1 Introdução

Como visto na figura 4, podemos utilizar um coeficiente de peso w para realizar o auto ajuste individual de cada elemento irradiante em um arranjo de maneira adaptativa, ou seja, realçando em tempo real as características desejadas para este canal. Este ajuste é feito por intermédio da utilização de algoritmos adaptativos, que se tornam cada vez mais comuns devido aos avanços no campo teórico na área de processamento de sinais.

Os filtros adaptativos buscam resolver problemas com variáveis desconhecidas ou problemas para os quais os filtros invariantes no tempo não são suficientes para solucioná-los (DINIZ, 2008). Através de um algoritmo e utilizando um sinal desejado como referência, estes filtros realizam o ajuste dos seus coeficientes de peso até que se atinja o sinal desejado de acordo com o critério do projetista.

Há diversos tipos de estruturas possíveis de serem implementadas para realizar estes ajustes, que podem ser sumariamente divididas em duas classes: filtros de resposta ao impulso finito (*Finite impulse response*, FIR) e filtros de resposta ao impulso infinito (*Infinite Impulse Response*, IIR). Neste projeto, trabalharemos com um filtro do tipo FIR, visto que a sequência de amostras de símbolos piloto enviada é finita, buscando atingir o ponto de convergência onde o sinal produzido é o mais próximo possível do sinal desejado, por meio da redução do erro quadrático médio, para os casos do NMLS e do RLS.

Os algoritmos escolhidos para ajustar os coeficientes de peso foram o *Normalized Least Mean Squares*(NLMS), uma variação do algoritmo mais utilizado para realização de filtragem adaptativa devido à sua simplicidade computacional, o *Least Mean Squares*(LMS), e o *Recursive Least squares*(RLS), pois, apesar de sua maior complexidade computacional, apresenta um menor tempo de convergência.

Para melhor compreender o processo de filtragem adaptativa e o funcionamento dos algoritmos propostos, será descrita, brevemente, a utilização do filtro de Wiener, uma vez que este serve de base para o desenvolvimento do LMS, RLS e suas variações.

3.2 Filtro de Wiener

Filtros adaptativos podem, de maneira geral, ser descritos como uma aproximação do Filtro de Wiener.

O filtro de Wiener é um filtro digital que tem, por objetivo, a restauração de sinal com ruídos. Para isso, ele utiliza a informação estatística do sinal original e da amostra ruidosa, estimando a relação entre ambas e, em seguida, aplica essa relação para filtrar o ruído e recuperar o sinal original.

A solução proposta por Wiener para realizar essa filtragem consiste na utilização do chamado vetor de correlação cruzada, \mathbf{p} , que estima a relação entre o sinal original e a amostra ruidosa, e da matriz de auto-correlação \mathbf{R} , responsável por estimar a variância do sinal original e a variância (DINIZ, 2008). As Equações para ambos os casos podem ser vistas em 3.1 e 3.2.

$$\mathbf{p} = E[d(k)\mathbf{x}(k)] \quad (3.1)$$

$$\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}(k)^T] \quad (3.2)$$

Neste caso, d é o sinal original e \mathbf{x} a amostra ruidosa do sinal. Wiener utiliza ambas as equações, para gerar os coeficientes de sua solução (DINIZ, 2008), que pode ser dada por:

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p} \quad (3.3)$$

Esta é a solução proposta por Wiener. Teoricamente, as informações dos sinais do sistema devem ser obtidas a priori para composição dessas mesmas variáveis. Na prática, porém, as estimativas precisas de \mathbf{R} e \mathbf{p} , não estão disponíveis. Contudo, a solução de Wiener serve de parâmetro de referência para demais algoritmos adaptativos implementados em cenários estacionários.

3.3 Algoritmos Adaptativos

3.3.1 Introdução

Para um sistema adaptativo linear, visto na Figura 6, a saída $\mathbf{y}(n)$ é uma estimativa do sinal desejado $\mathbf{d}(n)$. A diferença entre o sinal de referência e o sinal de saída gera um sinal de erro $\mathbf{e}(n)$, utilizado para ajuste dos coeficientes \mathbf{w} . Idealmente, uma vez que os valores de \mathbf{w} sejam ajustados ao ponto do sinal de saída ser igual ao sinal de referência, o valor de erro deve ser nulo.

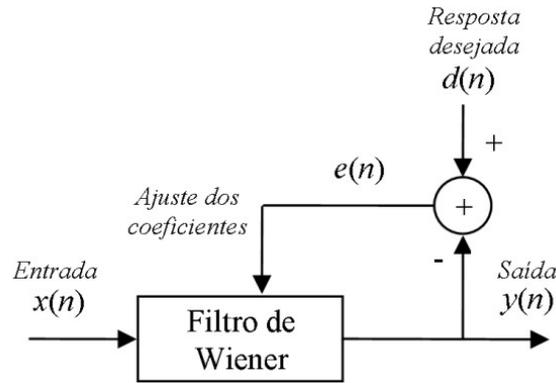


Figura 6 – Combinador linear, retirado de (NETO; GOMES, 2010)

Considerando que o filtro adaptativo consiste em um combinador linear, o sinal obtido é uma composição dos sinais combinados que vem de um vetor, e pode ser dada por:

$$\mathbf{y}(n) = \mathbf{w}^T(n)\mathbf{x}(n) \quad (3.4)$$

Adotando o critério de minimização da função custo do mínimo erro quadrático (*Minimum Square Error*, MSE) (DINIZ, 2008), e considerando que, para o caso complexo, ao invés da transposta deve-se obter a hermitiana do sinal, obtém-se as seguintes equações para o erro de estimação:

$$e(n) = d(n) - y(n) \quad (3.5)$$

$$\mathbf{e}(n) = d(n) - \mathbf{w}^H \cdot \mathbf{x}(n) \quad (3.6)$$

Essas Equações são o princípio básico do funcionamento de filtros adaptativos.

3.3.2 Algoritmo LMS

O algoritmo *Least Mean Square* (LMS) é amplamente utilizado para realizar filtragem adaptativa devido a sua simplicidade computacional, capacidade de convergência comprovada em cenários estacionários e comportamento estável nas mudanças do sinal de entrada (DINIZ, 2008). Seus parâmetros de filtro são considerados variantes no tempo, que o permite se acomodar a eventuais mudanças estatísticas do canal.

Em suma, o LMS, realiza um processo de filtragem, através do qual obtém uma saída, proveniente de um filtro discreto, e, posteriormente, a compara com a saída desejada, gerando assim, um sinal de erro estimado. Após isso, realiza o processo de adaptação,

ajustando os coeficientes de peso do filtro de maneira automática a partir do sinal de entrada e do erro estimado (NAHUZ, 2016).

Além disso, o cálculo do gradiente proposto pelo método de *steepest descent* (DINIZ, 2008) é substituído por meio de uma estimativa do mesmo, o que resulta em menor complexidade computacional. Contudo, isso gera um erro ou ruído de estimação, visto que a estimativa do gradiente é baseada em uma amostra única do vetor de entrada. Entretanto, o ruído diminui gradualmente ao passar do tempo, devido ao maior número de amostras obtidas.

Para encontrar o gradiente utilizado para ajuste dos coeficientes no algoritmo LMS, basta elevarmos a Equação 3.6 ao quadrado, obtendo, assim, uma equação de erro quadrático em forma vetorial:

$$[e(n)]^2 = [d(n)]^2 - 2d(n)\mathbf{x}^T \mathbf{w} + \mathbf{w}^T \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w} \quad (3.7)$$

Derivando-a, então, em relação a \mathbf{w} , obtemos:

$$\nabla J(n) = -2d(n)\mathbf{x}(n) + 2\mathbf{x}(n)\mathbf{x}^T \mathbf{w} \quad (3.8)$$

Observa-se que o termo $-2\mathbf{x}(n)$ pode ser posto em evidência:

$$\nabla J(n) = -2\mathbf{x}(n)[d(n) - \mathbf{x}^T(n)\mathbf{w}] \quad (3.9)$$

Por fim, utilizando a Equação 3.6 para substituir o valor entre colchetes, a equação do gradiente encontrada é:

$$\nabla J(n) = -2\mathbf{x}(n)e(n) \quad (3.10)$$

A partir desse valor, a equação de ajuste dos coeficientes para o algoritmo LMS é:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu\mathbf{x}(n)e(n) \quad (3.11)$$

Como estabelecido, a constante μ representa o tamanho do passo de adaptação. Desta forma, é ela quem regula a velocidade de convergência e a estabilidade do processo adaptativo. Logo, é crucial que seu valor seja escolhido de maneira apropriada, haja vista um valor muito alto para μ resultar em instabilidade e em oscilações, e, um valor muito baixo, em baixa capacidade de adaptação às mudanças estatísticas do sinal de entrada, em cenários não-estacionários. Para garantir a estabilidade e a convergência, o valor de μ deve seguir o critério da seguinte Equação:

$$0 < \mu < \frac{1}{\lambda_{max}} \quad (3.12)$$

Onde λ é o maior autovalor da matriz de auto-correlação dos sinais de entrada. Entretanto, é possível ajustar esse valor realizando-se estimativas em sequência para simplificação dos cálculos.

3.3.3 Algoritmo NLMS

Existem diversas variações do algoritmo LMS que permitem melhorar distintos aspectos de sua implementação, como a velocidade de convergência dos coeficientes ou a redução do erro residual em regime estacionário.

O *Normalized Least Mean Squares* - NLMS, é uma dessas variações, cuja principal diferença é o fato de o passo de adaptação μ ser normalizado pelo sinal de entrada $\mathbf{x}(\mathbf{n})$. O objetivo com esta alteração é a redução do ruído presente no algoritmo ocasionado pela estimação do gradiente para grandes valores de sinal de entrada (MOON; STIRLING, 2000).

Assim, o passo de adaptação passa a ser dado por:

$$\mu = \frac{\mu_n}{\|\mathbf{x}(n)\|^2} \quad (3.13)$$

E ao substituir esse valor na Equação (3.11), encontra-se:

$$\mathbf{w}(k+1) = \mathbf{w}(n) + \frac{\mu_n}{\|\mathbf{x}(n)\|^2} \mathbf{x}(n)e(n) \quad (3.14)$$

O NLMS apresenta não apenas a vantagem de reduzir o erro residual, mas também um tempo de convergência menor quando comparado ao algoritmo tradicional. Por outro lado, como desvantagem, a utilização da norma propicia uma complexidade computacional maior. O diagrama de blocos que exemplifica o funcionamento deste código pode ser visto em 7.

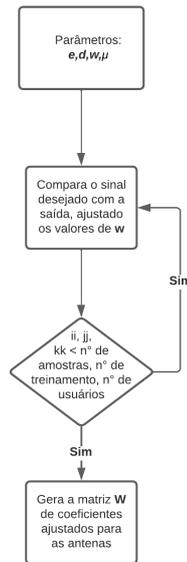


Figura 7 – Diagrama de blocos LMS

3.4 Algoritmo RLS

Como dito anteriormente, os algoritmos chamados de *least squares* (LS) buscam alcançar o ponto de convergência através da redução da soma quadrática da diferença entre o sinal desejado e o sinal obtido (DINIZ, 2008). Para o LMS e NLMS, descritos nas subseções anteriores, o método utilizado para alcançar o ponto de convergência consiste em reduzir o erro quadrático médio a partir da comparação entre o sinal desejado e o sinal obtido. Porém, outra área de estudos de filtragem adaptativa busca utilizar as amostras do sinal obtidas a cada iteração para computar os dados de maneira recursiva, formando assim o *recursive least squares* (RLS).

Como também é derivado da solução ótima do filtro de *Wiener*, a base de funcionamento deste algoritmo é similar a descrita para o LMS/NLMS. Assim, a Equação 3.6 pode ser utilizada para ajuste do erro também neste algoritmo.

Quanto ao ajuste dos pesos, para o caso dos algoritmos de LS, a função objetiva para ajuste é dada por:

$$\sum_{i=0}^k \mu^{k-i} [d(i) - x^T(i)w(k)]^2 \quad (3.15)$$

Nessa equação, a variável μ representa o chamado de fator de esquecimento, uma vez que as informações obtidas previamente tem cada vez menos importância na atualização dos coeficientes de peso. Diferenciando a Equação 3.15 em relação a w e igualando o resultado a zero, obtém-se a equação ótima de ajuste dos coeficientes de peso, expressa por:

$$\mathbf{w}(k) = \left[\sum_{i=0}^k \mu^{k-i} x(i) x^T(i) \right]^{-1} \sum_{i=0}^k \mu^{k-i} \mathbf{x}(i) d(i) \quad (3.16)$$

$$= \mathbf{R}_D^{-1}(k) \mathbf{p}_D(k) \quad (3.17)$$

$\mathbf{R}_D(k)$ e $\mathbf{p}_D(k)$ são, respectivamente, a matriz determinística de correlação do sinal de entrada e o vetor determinístico de relação cruzada entre o sinal de entrada e o sinal desejado. Considerando o lema de inversão de matrizes (WOODBURY; STATISTICS, 1950), dado por:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C + DA^{-1}B)^{-1}DA^{-1} \quad (3.18)$$

Temos que, para o caso complexo:

$$\mathbf{S}_D(k) = \mathbf{R}_D^{-1}(k) = \frac{1}{\mu} \left[S_D(k-1) - \frac{S_D(k-1) \mathbf{x}(k) \mathbf{x}^H(k) S_D(k-1)}{\mu + \mathbf{x}^H(k) S_D(k-1) \mathbf{x}(k)} \right] \quad (3.19)$$

Assim, obtém-se que o algoritmo complexo para RLS é:

$$e(k) = d(k) - \mathbf{w}^H(k-1) x(k)$$

$$\psi(k) = S_D(k-1) x(k)$$

$$\mathbf{S}_D(k) = \frac{1}{\mu} \left[S_D(k-1) - \frac{\psi(k) \psi^H(k)}{\mu + \psi^H(k) \mathbf{x}(k)} \right]$$

$$\mathbf{w}(k) = \mathbf{w}(k-1) + e^*(k) \mathbf{S}_D(k) \mathbf{x}(k) \quad (3.20)$$

O algoritmo RLS apresenta maior complexidade computacional quando comparado ao NLMS. Porém, devido ao fator de recursividade, ele apresenta um tempo de convergência mais rápido e grande estabilidade às variações no sinal de entrada. Ambos os algoritmos foram implementados como classes no simulador, e utilizados para realização das simulações. A Figura 8 exemplifica o código através de diagrama de blocos.

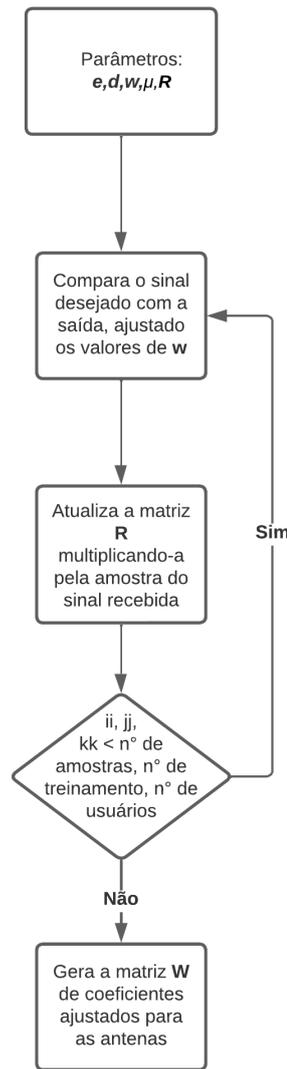


Figura 8 – Diagrama de blocos RLS

3.5 Algoritmo MRT

O *Maximum Ratio Transmission* (MRT) é um algoritmo utilizado na otimização de sistemas de comunicações sem fio para melhorar a qualidade do sinal recebido. Trata-se de uma técnica de multiplexação espacial que permite a maximização da relação sinal-ruído (SNR) no receptor.

O MRT é comumente utilizada em sistemas MIMO para transmissão do sinal. Nela, o sinal transmitido por cada antena é direcionado individualmente para o receptor e, em seguida, o receptor combina o sinal recebido de cada antena de tal forma que a SNR seja maximizada (CHAITANYA; REDDY, 2020).

Considerando que o canal \mathbf{H} foi previamente estimado através de um filtro no tempo, os coeficientes de peso \mathbf{W} são dados por:

$$\mathbf{W} = \mathbf{H}^H \quad (3.21)$$

A principal vantagem do MRT é que ele permite a maximização da capacidade do sistema, aumentando a taxa de dados e melhorando a qualidade do sinal. Além disso, é uma técnica simples e eficiente que não requer conhecimento prévio do canal de propagação.

Em contrapartida, comparado aos algoritmos NLMS e RLS, o MRT realiza uma estimação do canal durante o processo de conformação de feixe, o que aumenta a complexidade computacional e acaba influenciando no direcionamento do feixe.

3.6 Algoritmo proposto

Dois algoritmos distintos foram elaborados para realização deste trabalho. O primeiro, para o cenário simplificado, destina-se à validação do funcionamento dos algoritmos, no processo de conformação de feixes. Após a validação, os algoritmos foram implementados no simulador de canais de rádio *Hermespy*, no qual, diversas de suas funcionalidades já implementadas foram utilizadas para realização das simulações. Este capítulo explicará o funcionamento lógico de ambos os códigos

3.6.1 Algoritmo simplificado

Este trabalho destina-se a implementar os algoritmos supracitados no simulador de canal *Hermespy* como opções para realizar o processo de conformação de feixes e, após a implementação, avaliar a performance destes no simulador. A Equação que rege o funcionamento do NLMS é vista em 3.14, e a que rege o funcionamento do RLS em 3.20.

Os algoritmos foram previamente validados em um canal simples, no qual são utilizados como entrada uma série de parâmetros referentes ao sinal, por exemplo: o valor da frequência para cálculo do comprimento de onda do sinal λ ; o ângulo que indica a *DOA* do sinal em linha de visada, θ ; a variação sofrida em graus para o segundo percurso percorrido pelo sinal, $\Delta\theta$; a quantidade de antenas receptoras presentes na ERB n_{ant} ; a sequência de símbolos pilotos de tamanho n_{amos} ; o valor em dB para o fator de Rice K ; o valor em dB para a relação sinal-ruído *SNR* e o tamanho do passo de adaptação μ . A partir destes, o programa realiza o processo de conformação de feixe na direção do usuário, a partir do ajuste iterativo dos coeficientes \mathbf{w} . Um resumo do funcionamento do algoritmo proposto pode ser visto logo abaixo, em forma de digrama de blocos.

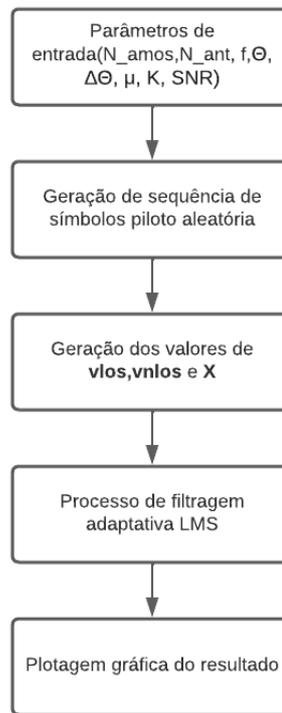


Figura 9 – Diagrama de blocos do algoritmo proposto

3.6.2 Algoritmo *Hermespy*

O código feito para simular o processo de *beamforming* no *Hermespy* foi estruturado utilizando o paradigma de orientação a objetos. Dessa forma, temos classes que representam partes do processo de conformação de feixe e da realização da simulação, e cada uma delas possui atributos e métodos utilizados para construir a simulação final. O diagrama de classes UML, a representação mais comum para códigos orientados a objetos, pode ser visto em [11](#). Cada uma das classes terá sua função, seus atributos e métodos explicados ao longo desta seção.

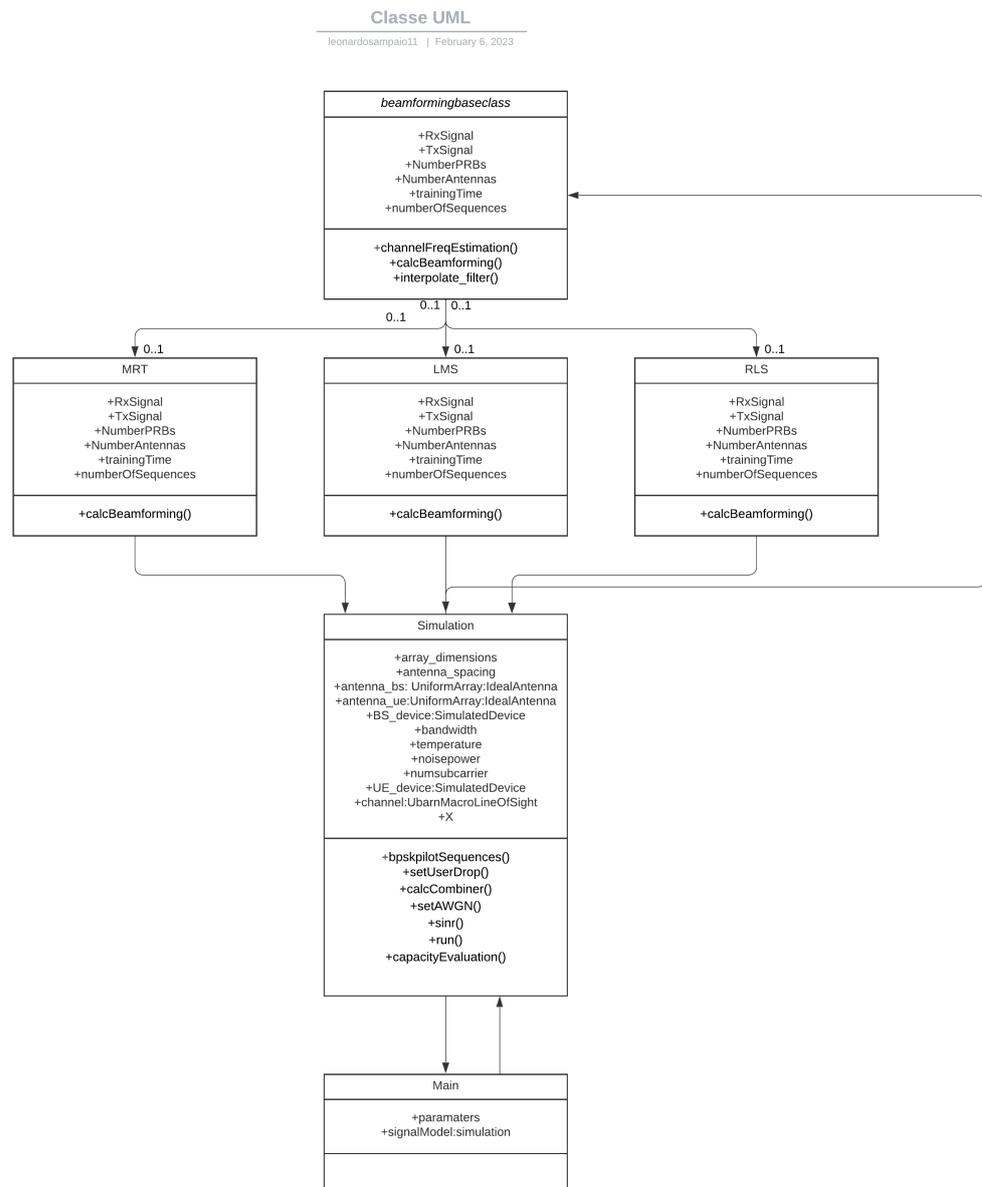


Figura 10 – Diagrama de classes do algoritmo proposto

3.6.2.1 Ambiente principal

No ambiente principal de código, há a inicialização do mesmo, onde o usuário define os parâmetros de simulação. Estes incluem: a frequência da portadora; número de PRBs totais e de treinamento; espaçamento da subportadora; número de antenas nos eixos x e y ; número de símbolos OFDM; tipo de cenário; algoritmo para formação do feixe; número de usuários; variação máxima do ângulo para os eixos x e y ; potência do sinal e potência de treinamento. Esses valores serão passados para a classe *simulation* através de um dicionário usando, utilizando os chamados *KeyWords Arguments*, ou *kwargs*. Após a passagem desses parâmetros, o código chama o método *capacityEvaluation*, também

da classe *simulation* para obter os valores de SINR, eficiência espectral, capacidade do sistema e capacidade do sistema por usuário.

3.6.2.2 Classe *simulation*

A classe *simulation* é responsável por gerar os valores da simulação a partir dos parâmetros recebidos do ambiente principal. O simulador *Hermespy* possui uma série de funções implementadas para geração dos elementos do canal, e várias serão utilizadas para a simulação.

Primeiramente, são utilizadas as funções de *UniformArray* junto com a de *IdealAntenna*, para configuração do arranjo de antenas. Essas funções recebem como parâmetro a quantidade total de antenas e o espaçamento entre elas. Em sequência, é gerada a ERB simulada, com as informações de frequência da portadora, número de antenas, posição e orientação.

O código define, então, uma série de propriedades para classe. São elas: a largura de banda do canal, o número de subportadoras, a temperatura do ambiente e a potência de ruído.

Em sequência, os métodos de *bpskpilotSequences*, quando chamados, geram a sequência de símbolos piloto de acordo com a modulação binária por deslocamento de fase, o *setUserDrop*, que aloca o usuário aleatoriamente no sistema de coordenada cartesiano nos eixos x e y , porém, mantém o eixo z fixado em 1,5, *setAWGN*, que gera o sinal de ruído a partir da potência de ruído e da forma do sinal transmitido, *run* e *capacityEvaluation*.

O método *run* possui uma variedade de objetivos, como: construir os dispositivos dos usuários com as informações do número de antenas, frequência da portadora, posição e orientação; chamar a função de geração dos símbolos pilotos; definir os PRBs que serão utilizados para treinamento; chamar a função *UrbanMacrocellsLineOfSight*, que, como dito anteriormente, estabelece o cenário de simulação como macro urbano em linha de visada, a partir dos parâmetros de UE e ERB criados; gera e propaga o sinal através da função *signal* e do método *propagate*, ambas funções nativas do simulador; gera a matriz do canal e retorna os valores da matriz de sinais do canal já com ruído AWGN, a sequência de símbolos piloto e a posição do usuário.

O método de *capacityEvaluation* é chamado a partir do ambiente principal logo após o retorno dos valores do método *run*. Uma vez recebidos o sinal, as amostras de símbolo piloto e a posição do(s) usuário(s), este método realiza a chamada do método *calcBeamforming* que utiliza a classe correspondente ao algoritmo escolhido para o processo de conformação de feixe. Com o valor retornado de \mathbf{W} , estima a matriz de interferência \mathbf{I} e da relação SINR. Por fim, calcula os valores da eficiência espectral, capacidade do sistema e capacidade do sistema por usuário, retornando-os para o ambiente principal.

Percebe-se que este método é um dos principais responsáveis por gerar as informações necessárias do canal para realização da simulação, e que o simulador *Hemespy* possui diversas funções já implementadas para geração dos mesmos, e com a habilidade de gerar representações mais próximas da realidade.

3.6.2.3 Classe *beamformingbaseclass*

A *beamformingbaseclass* configura-se como uma superclasse, pois serve de base para inicialização das demais classes de geração de feixes MRT, RLS, LMS, que herdam suas características.

São iniciados os valores de *RxSignal* e *TxSignal* que representam, respectivamente, o sinal recebido pela ERB e as amostras de símbolo piloto enviadas. A partir das informações do formato dessas duas matrizes são geradas as propriedades de número de PRBs de treinamento, número de antenas, quantidade de símbolos OFDM e número de usuários. Logo após, o método *channelFreqEstimation* realiza a estimação do canal para o posterior cálculo dos coeficientes \mathbf{W} para o algoritmo MRT. Vale ressaltar que essa estimação não é necessária para o LMS e o RLS. Por último, nesta classe, há o método de *calcBeamforming*, que é apenas inicializado aqui para ser herdado pelas demais classes, e o método *interpolate_filter*, que realiza a interpolação dos pesos \mathbf{W} estimados com as amostras de treinamento para as demais amostras enviadas. O programa conta com a interpolação do tipo *Spline*, cubica ou linear.

As classes de MRT, LMS E RLS herdam os parâmetros e propriedades desta classe, realizam as estimativas dos coeficientes de ajuste baseado nos algoritmos descritos nas subseções 3.3, 3.4 e 3.5, e chamam o método *interpolate_filter*. Assim, obtém-se o valor dos pesos para todas as amostras, que são então retornados para o método *capacityEvaluation* da classe *Simulation*.

O diagrama de blocos que exemplifica o funcionamento lógico deste código pode ser visto em:

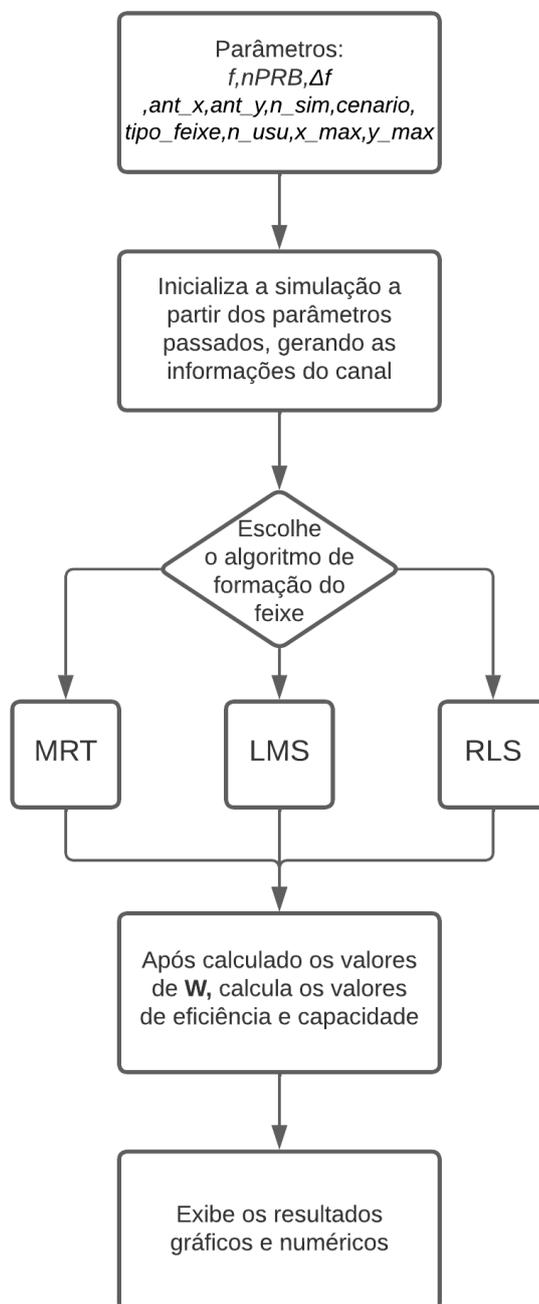


Figura 11 – Diagrama de blocos do algoritmo proposto

4 Simulações e resultados

A partir do que foi desenvolvido ao longo deste trabalho, neste capítulo são realizadas as simulações, variando diferentes parâmetros do código nos cenários descritos no capítulo 2.

4.1 Simulações simplificadas

Antes de realizar a implementação dos blocos de código no simulador, foram realizadas simulações no cenário simplificado para validar o funcionamento dos mesmos.

Para isso, alguns parâmetros de entrada foram arbitrariamente escolhidos e mantidos fixos ao longo de todas as simulações para que, quando se variasse os demais parâmetros, as observações pudessem ser devidamente feitas sob mesma ótica. Tais parâmetros são: a frequência da portadora escolhida, utilizada para cálculo do comprimento de onda, de 3.5 Ghz, principal bloco do leilão 5G realizado recentemente no Brasil; o ângulo em que se encontra o usuário, estabelecido em 75° , que representa a direção na qual o feixe deverá ser apontado; o tamanho do passo de convergência, de 0,01, escolhido após a realização de uma série de simulações e observar sua convergência para vários dos casos apresentados.

Os demais valores de entrada foram alterados alternativamente, ou seja, enquanto um deles era variado, os outros permaneciam em valores fixos. Destarte, os valores fixos estabelecidos foram de: 64 para o número de antenas; 5dB para a relação sinal-ruído; 1dB para o fator de Rice; 100 para o tamanho da sequência de treinamento de símbolos pilotos. Ressalta-se que a SNR presente neste canal é simplificada e manualmente inserida para gerar o vetor de ruído, e, por isso, não se assemelha a mesma existente em canais de rádios realísticos.

Cada simulação é composta por 4 iterações, que tem seus valores de erro quadrático e o *beamforming* resultante plotados simultaneamente e identificados por meio de cores para posterior comparação. A tabela 2 apresenta os valores mantidos fixos, e a 3 apresenta os parâmetros variados e o valor para cada iteração.

Parâmetros	Valor
Frequência da portadora	3.5 GHz
Posição do usuário	75°
μ nlms	0,01
μ rls	0,99

Tabela 2 – Parâmetros fixos para a simulação simplificada

Parâmetros	1 ^a iteração	2 ^a iteração	3 ^a iteração	4 ^a iteração
Símbolos piloto	25	50	75	100
SNR [dB]	-20	-10	1	10
Rice [dB]	1	5	10	15
Antenas	8	16	32	64

Tabela 3 – Parâmetros Variados para a simulação simplificada

4.1.1 Variação do número de símbolos pilotos

O primeiro parâmetro variado foi o tamanho da sequência de símbolos pilotos enviados. Para tal, foram feitas 4 iterações variando os valores em 25, 50, 75 e 100.

Apesar de apenas 4 valores serem considerados, a simulação foi repetida diversas vezes, a fim de se observar os possíveis efeitos causados pela presença de ruído AWGN, bem como pela seleção aleatória dos símbolos piloto. Desta forma, os resultados apresentados em 12 e 13 explicitam uma das versões testadas, que possuíam características bem semelhantes entre si, variando geralmente apenas o tempo necessário para convergência e a ganho obtido.

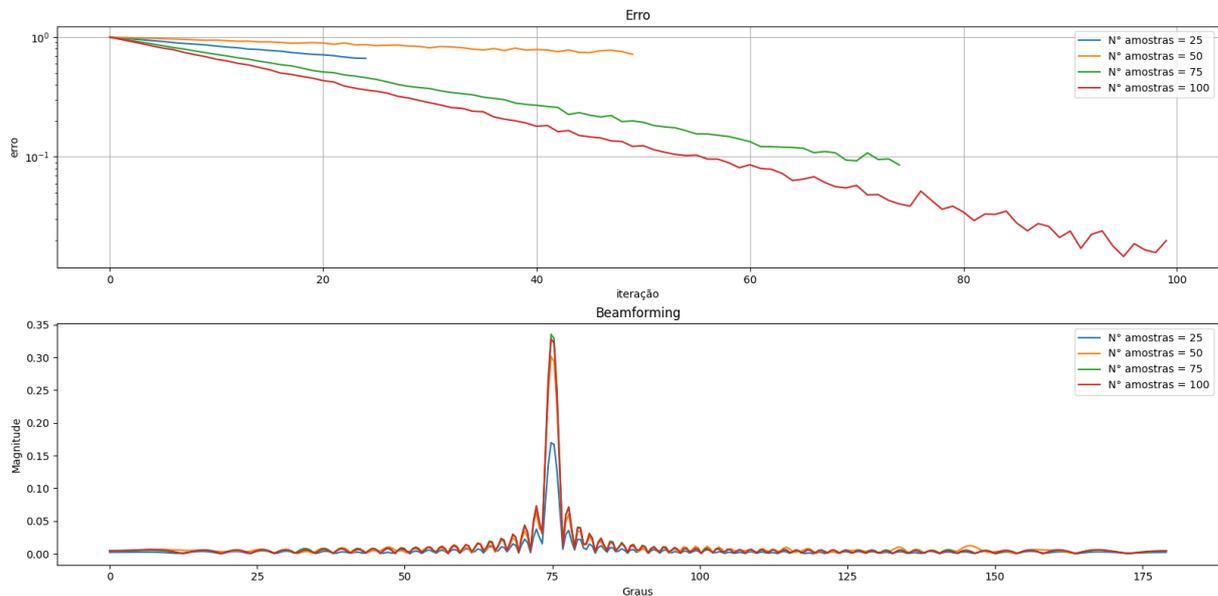


Figura 12 – Resultado para variação na quantidade de amostras de símbolos pilotos para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

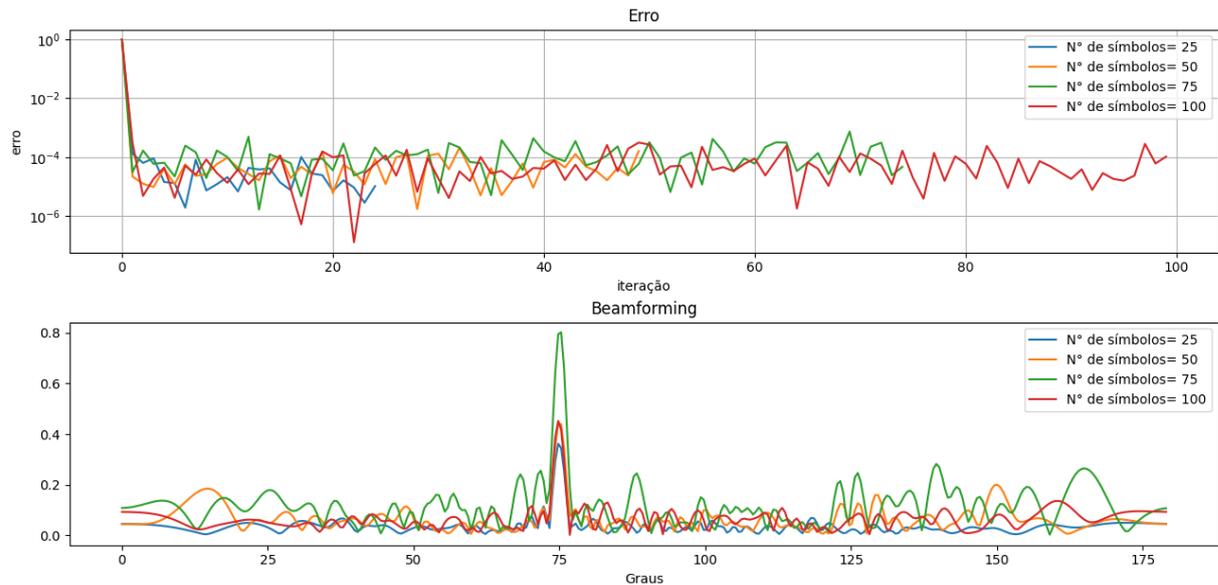


Figura 13 – Resultado para variação na quantidade de amostras de símbolos pilotos para o RLS, onde: (a) Representa o valor de erro quadrático; (b) representa o resultado do processo de conformação de feixe

O resultado das primeiras simulações nos mostra que é possível atingir o ponto de convergência com diferentes sequências de símbolos pilotos, e que, não necessariamente, mais quantidade representa um maior ganho de *beamforming*, pois, ao gerar símbolos pilotos aleatoriamente, a variação estatística pode atrasar o processo de convergência do mesmo, além do fato de quanto maior a sequência de amostra de símbolos piloto, maior é o tempo de aquisição de CSI, resultando em um dos maiores problemas enfrentados em sistemas MIMO chamado de *overhead*, e que é o principal limitador de usuários por célula.

Há estudos que buscam gerar estes símbolos de maneira ótima (SO et al., 2015) (NOH et al., 2014), fazendo com que os símbolos enviados possuam as melhores estimativas do canal ou, ainda, incrementem aspectos como a relação sinal-ruído.

4.1.2 Variação na relação sinal-ruído

O segundo parâmetro variado foi o valor da SNR, inicialmente, em modo de varredura, a fim de se encontrar o menor valor possível para ele, e que permitisse a formação de um feixe. O valor encontrado foi de aproximadamente -20dB. A partir dele, foram realizadas 4 interações, acrescentando o valor sempre na ordem de 10 dB, ou seja, -20, -10, 1 e 10 dB. Os resultados presentes em 14 15 mostram duas das simulações feitas para observação deste parâmetro

O resultado para o LMS exibido na Figura 14 mostra que, para todas as variações, o erro convergiu e o feixe foi formado. Já para o RLS, visto na Figura 15, para os valores

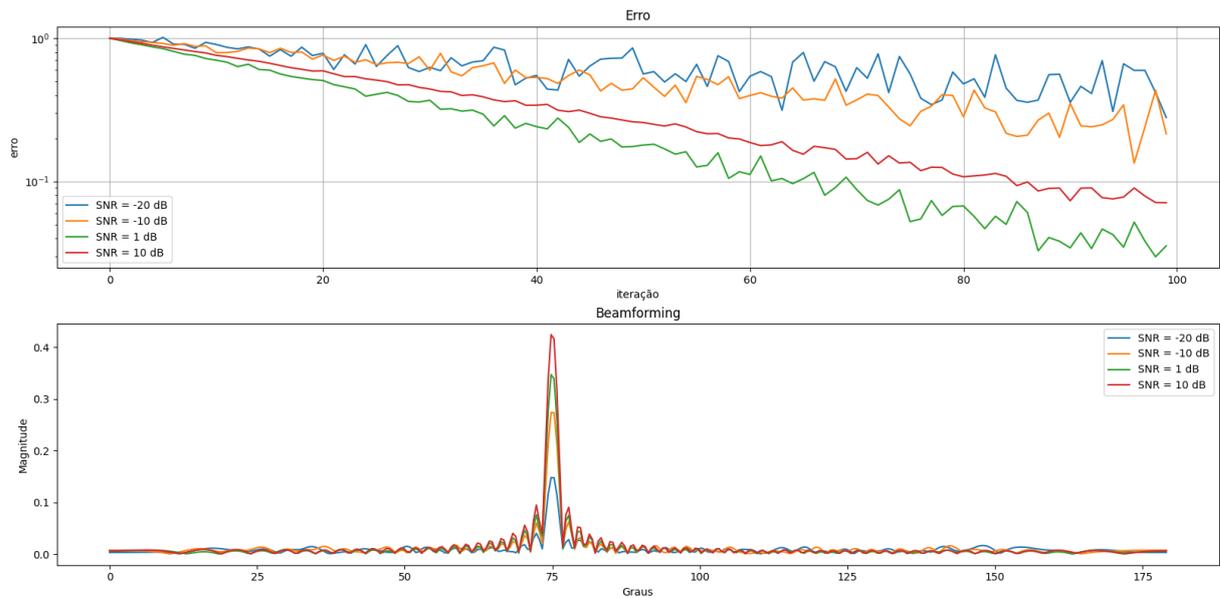


Figura 14 – Resultado para variação no valor de SNR para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

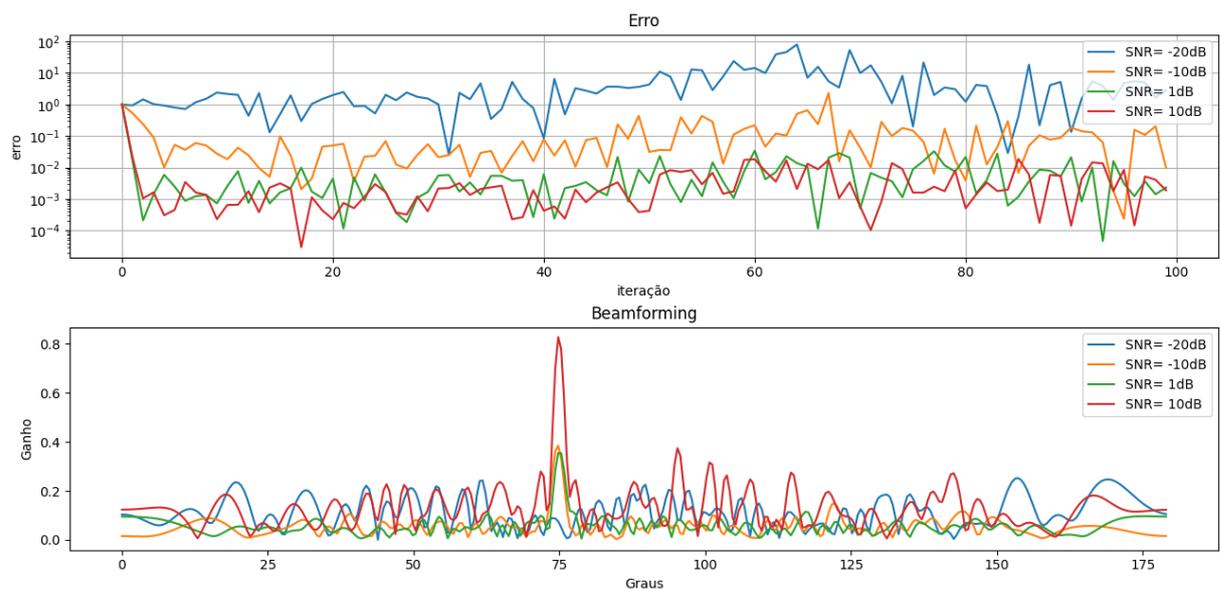


Figura 15 – Resultado para variação no valor de SNR para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

mais baixos de SNR o erro não converge e o feixe não é simulado. Isto pode ter sido ocasionado pela escolha do valor para o fator de esquecimento ou para o erro a priori. Como discutido anteriormente, valores muito baixos para passo de adaptação para o NLMS resultam em baixa capacidade de adaptação as mudanças do sinal de entrada, então, o valor aqui foi escolhido acertadamente.

No entanto, as simulações demonstram a grande capacidade de performance do

NLMS e do RLS em cenários com uma má relação de SNR, visto que, para -10dB, o código já se apresentou de maneira estável. Assim, em locais onde há presença de muito ruído, pode-se utilizar o NLMS e o RLS para redução de interferência de sinais indesejados. Para melhor aproveitamento do RLS neste cenário, outros valores podem ser adotados para o fator de esquecimento e o erro a priori.

4.1.3 Variação no fator de Rice

O terceiro parâmetro variado foi o valor do fator Rice, que implica diretamente na potência da componente direta V_{los} sobre as demais componentes V_{nlos} . Isto, pois o parâmetro k relaciona a potência da componente dominante com a potência das demais componentes do percurso. Com isso, a variação de k para um valor mais alto tende a tornar o canal empobrecido, posto que a diversidade de percursos é perdida, e o mesmo tende a uma distribuição normal (RODRIGUES, 2003). Os valores escolhidos foram de 1, 5, 10 e 15 dB. Os resultados obtidos podem ser vistos nas Figuras 16 e 17. Vale ressaltar que é possível realizar o cálculo da estimativa ótima para o valor do parâmetro K através de cálculos computacionais, contudo, estes, por vezes, são de alta complexidade (ABDI et al., 2001). Os demais tipos de desvanecimento, como o de Rayleigh e o Nakagami, não foram abordados neste trabalho devido ao cenário proposto de linha de visada.

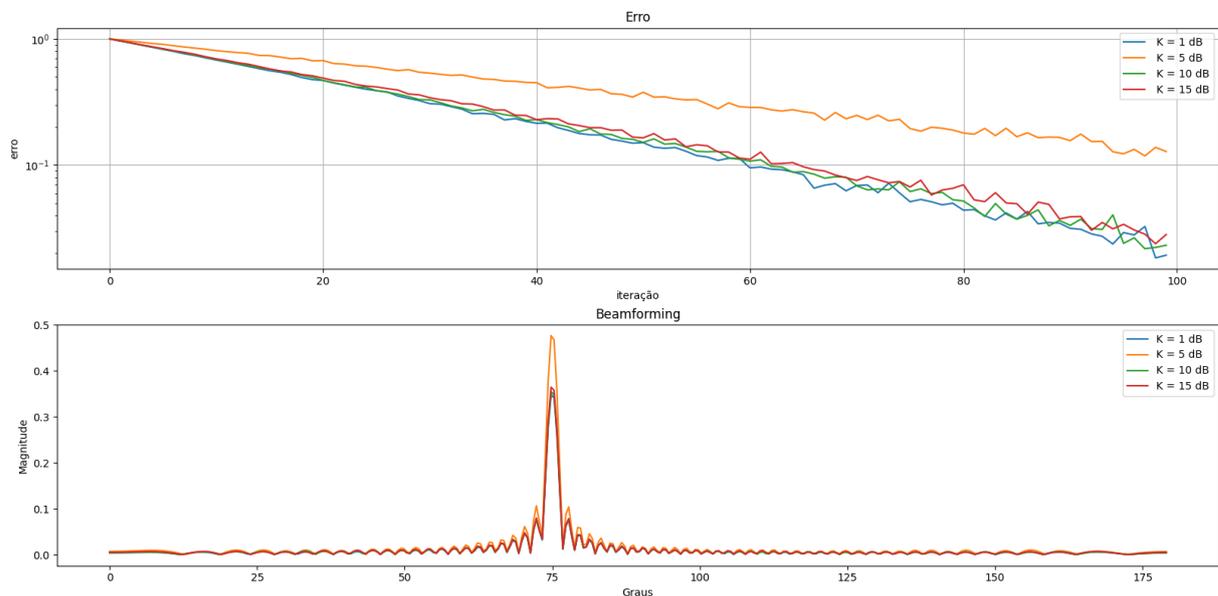


Figura 16 – Resultado para variação no valor de k para o NLMS, onde: (a) Representa o valor de erro quadrático; (b) representa o resultado do processo de conformação de feixe

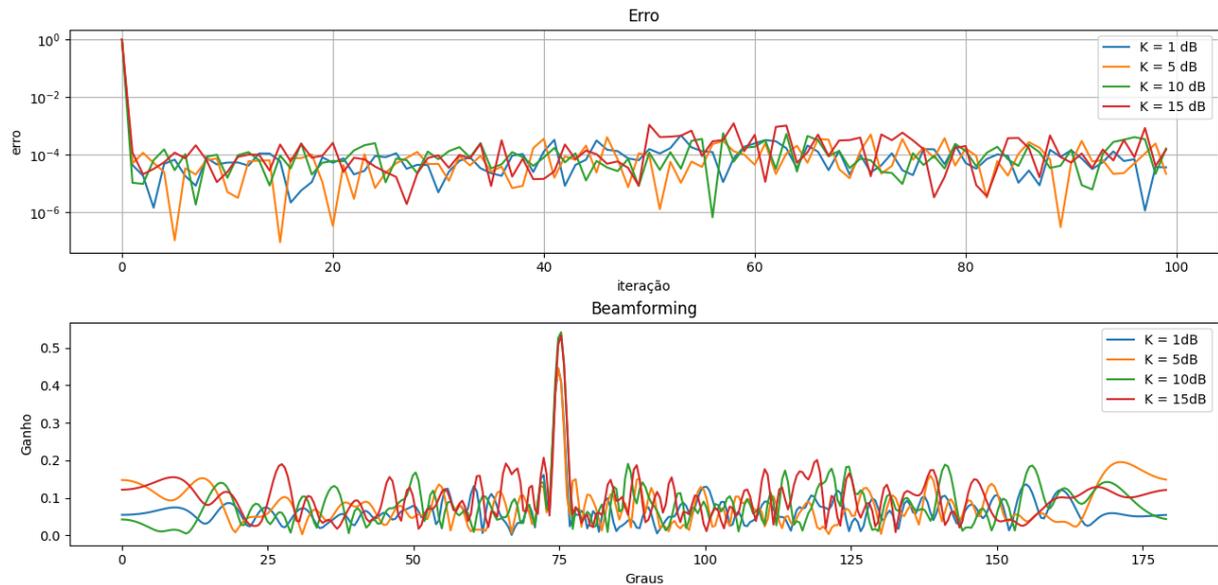


Figura 17 – Resultado para variação no valor de k para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

É possível observar que não há grandes variações no quesito de erro e no quesito de formação de feixes. As mudanças na ordem de ganho e de tempo de convergência são proporcionadas apenas pela variação de ruído.

4.1.4 Variação no número de antenas

O quarto parâmetro a ser variado foi o número de antenas utilizadas para simulação. Os valores escolhidos para simulação foram de 8, 16, 32 e 64 antenas, e foram baseados em valores de equipamentos de transmissão utilizados em enlaces comerciais, como as *Active Antenna Units* (AAUs), da fabricante Huawei, que possuem versões com 8, 32 e 64 antenas transmissoras e receptoras (HUAWEI, 2022). Os estudos mais recentes contam com a utilização de 128 antenas, porém, ainda se encontram em fases de testes. Os resultados podem ser vistos em 18 e 19.

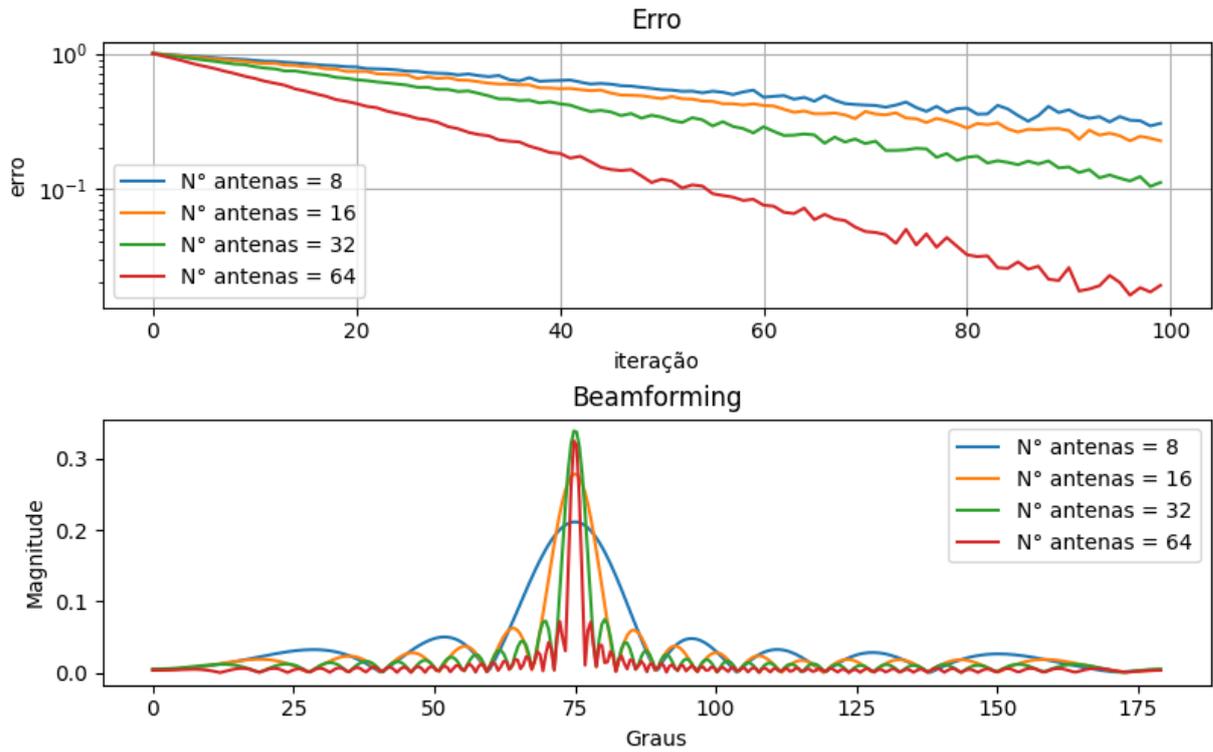


Figura 18 – Resultado para variação no número de antenas para o NLMS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

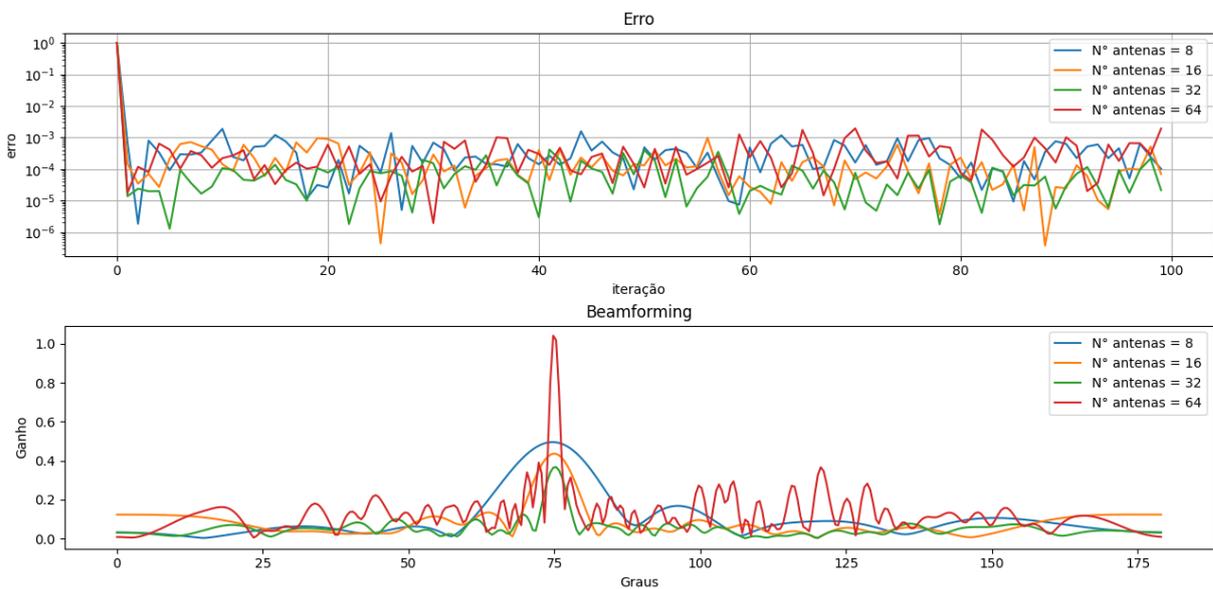


Figura 19 – Resultado para variação no número de antenas para o RLS, onde: (a) Representa o valor de erro quadrático;(b) representa o resultado do processo de conformação de feixe

Aqui, as visualizações de interesse são representadas nas Figuras 18b e 19b. Nestas, fica evidente a diferença entre o estreitamento do feixe para os diferentes números de antenas. Logo, a maior quantidade de antenas proporciona o maior ganho de multiplexação espacial, permitindo uma menor ocupação da banda do canal. O ganho de *beamforming* cresce linearmente com relação ao número de antenas, desde que se tenha conhecimento prévio das CSI. Na ausência dessas informações, o ganho, eventualmente, tende a um crescimento logarítmico (MARZETTA, 2015).

4.2 Simulações no *Hermespy*

Após a verificação do funcionamento dos códigos propostos no cenário simplificado, os códigos foram adaptados e integrados no simulador de canais *Hermespy*, e uma validação semelhante à anterior foi realizada, na qual foram mantidos alguns valores fixos de referência, enquanto outros são variados para verificar o desempenho dos algoritmos. Nesta etapa, além de avaliar o desempenho dos códigos implementados em um cenário mais realístico, o desempenho deles será comparado com o desempenho do algoritmo MRT já existente no simulador.

Assim, permanecerão fixos os valores: da frequência central da portadora, do espaçamento entre subportadoras, do passo de convergência para o NLMS, e do fator de esquecimento para o RLS, nos valores previamente estabelecidos na subseção anterior; as potências do sinal enviado e a potência da amostra de treino, em 40 dBm e 5 dBm, respectivamente; o número de símbolos OFDM em 14; o número de PRBs em 12.

Para os parâmetros variáveis, que também seguirão a lógica de alternância previamente estabelecidas, mantém-se o valor de 64 antenas, enquanto fixa-se o valor do espaçamento entre subportadoras em 15 KHz, o número de usuários em 1, o número de PRBs de treinamento em 2, e o desvio máximo para x e y ambos em 10. Vale ressaltar que os valores de espaçamento entre subportadoras seguirão a lógica apresentada na tabela 1.

Inicialmente, foram realizadas mil iterações para cada algoritmo, com um usuário alocado nos ângulos entre 80° a 85° graus polares e azimute de 45° graus, para que assim fosse possível realizar o cálculo da função de distribuição cumulativa (CDF) para a eficiência espectral e capacidade do sistema da área. As Figuras 20, 21 e 22 trazem os resultados gráficos para, respectivamente, os algoritmos de MRT, NLMS e RLS. Já a tabela 4 mostra os resultados obtidos numericamente para a probabilidade cumulativa de 0,5.

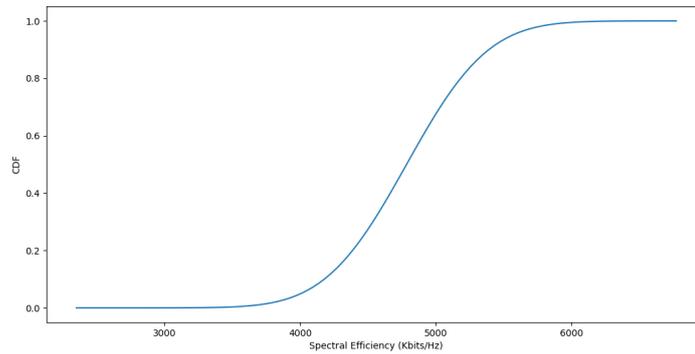


Figura 20 – Resultado para a CDF do NLMS

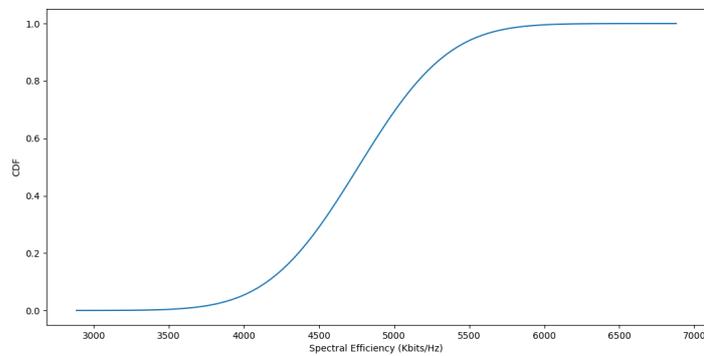


Figura 21 – Resultado para CDF do RLS

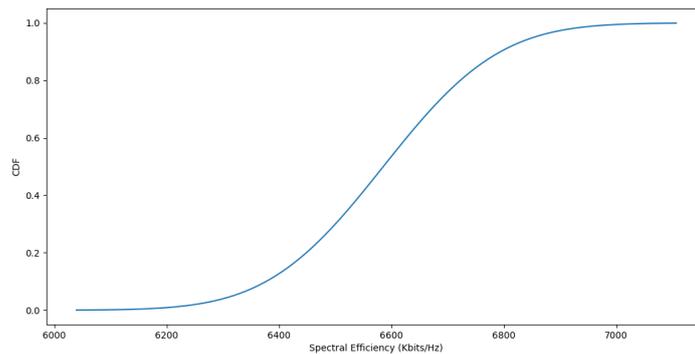


Figura 22 – Resultado para CDF do MRT

É possível perceber na tabela 4 que a capacidade do sistema e a eficiência espectral é maior para o MRT, e aproximadamente igual para o RLS e NLMS. Isso ocorre devido à estimação prévia da frequência do canal feita pelo MRT.

As simulações a seguir serão plotadas com apenas uma iteração e, ao contrário do cenário simplificado, onde as simulações foram realizadas bidimensionalmente devido ao

Algoritmo	Capacidade [Mbits/Hz]	Eficiência espectral[bits/Hz]
MRT	14200.14	6.57
RLS	10495.28	4.85
NLMS	10501.03	4.86

Tabela 4 – Valores de meio percentil de capacidade e eficiência espectral para mil iterações

fato das antenas estarem dispostas em um ULA, aqui, as simulações serão tridimensionais, pois elas se encontram dispostas em um URA, e para melhor visualização da posição de cada usuário e do ângulo de apontamento, com os usuários alocados em posições aleatórias numa variação de 180° graus. Todas as simulações terão seu *beamforming* resultante plotado, com algumas variações de rotação no gráfico para melhor visualização. Para o caso de variação no número de usuários, os valores de eficiência espectral, a capacidade do sistema, e a capacidade do sistema por usuário serão exibidos em tabelas. Esses valores não serão exibidos para as simulações com apenas um usuário, pois sempre se aproximam do exibido na Tabela 4. Assim, as tabelas 5 e 6 apresentam, respectivamente, os parâmetros variáveis e fixos das simulações.

Parâmetros	1ª iteração	2ª iteração
Potência [dBm]	5	20
Nº PRBs de treino	2	8
Desvio X e Y	10	20
Nº usuários	2	3

Tabela 5 – Parâmetros Variados para a simulação no *Hermespy*

Parâmetros	Valor
Frequência da portadora	3,5 GHz
Espaçamento da subportadora	15 kHz
Nº Antenas	64
Nº PRBs totais	12
Nº de símbolos OFDM	14
Potência de transmissão [dBm]	40
Cenário	Macro Urbano em Linha de Visada

Tabela 6 – Parâmetros fixos para a simulação no *Hermespy*

4.2.1 Variação na potência do sinal

O primeiro dos parâmetros modificados para verificação dos algoritmos implementados no simulador foi o de potência de treino. A potência de treinamento refere-se a uma parcela da potência do sinal total que será utilizada no treinamento para estimação do canal.

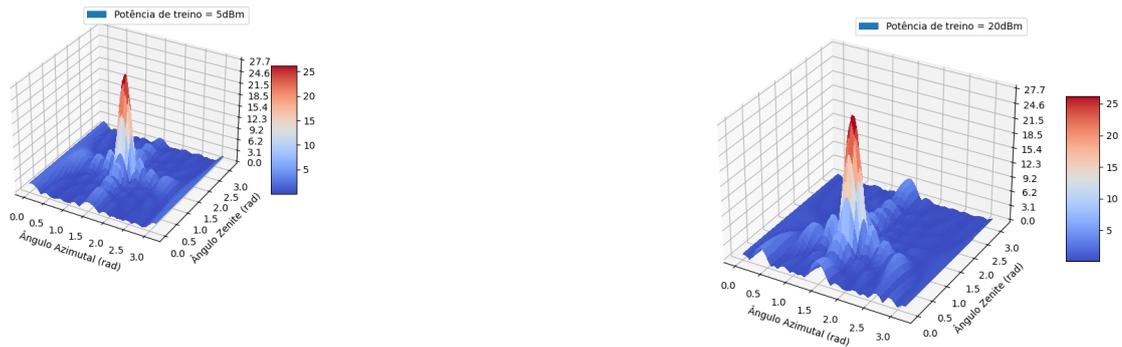


Figura 23 – Resultado para variação na potência de treinamento para o NLMS, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe

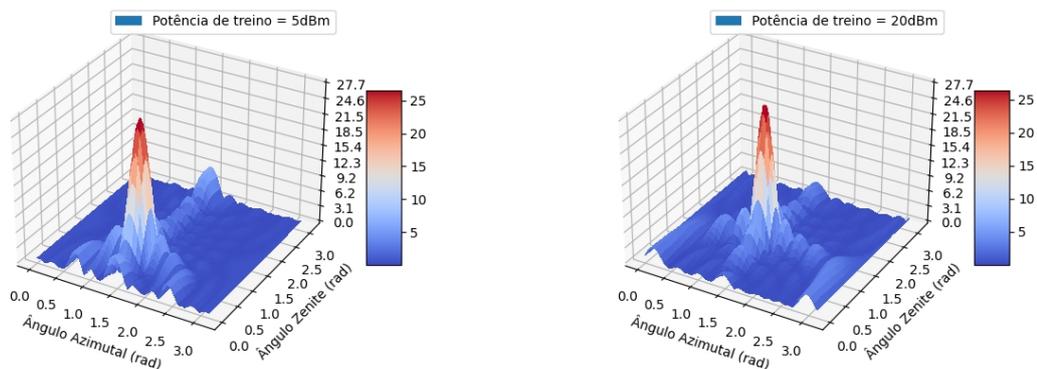


Figura 24 – Resultado para variação na potência de treinamento para o RLS, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe

Ao analisar os resultados obtidos, pode-se inferir que o aumento da potência não propicia ganhos evidentes na construção do feixe em nenhum dos casos. Desta forma, para o cenário proposto, uma pequena parcela da potência total do sinal enviado é o suficiente para realização da técnica de *beamforming* de maneira precisa.

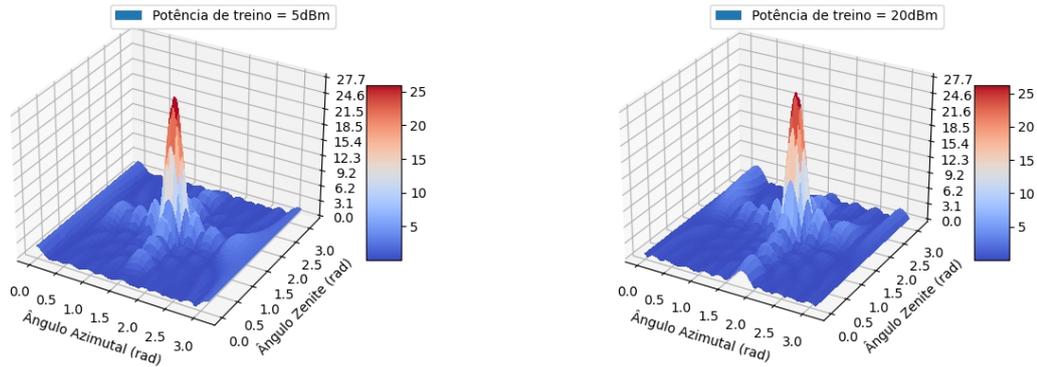


Figura 25 – Resultado para variação na potência de treinamento para o MRT, onde: (a) Representa o resultado para 5 dBm;(b) representa o resultado para 20 dBm conformação de feixe

4.2.2 Variação no número de PRBs de treinamento

O código utiliza as sequências de treinamento para realizar as estimativas iniciais dos pesos \mathbf{W} , e, posteriormente, realiza a interpolação dos valores de treino para estimar os pesos para os demais PRBs disponíveis. Inicialmente, o valor escolhido foi o de 2 PRBs de treinamento, cada um com 14 símbolos piloto OFDM. Os parâmetros serão variados de 2 para 8 amostras, a fim de verificar o impacto de um tamanho de treinamento maior para o processo de formação de feixes.

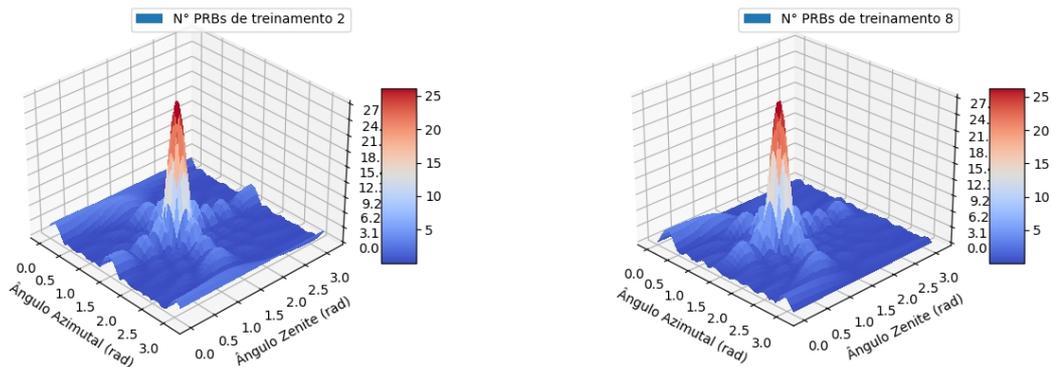


Figura 26 – Resultado para variação no número de amostras de treinamento RLS, onde: (a) apresenta o resultado para o valor de 2 amostras de treino;(b) apresenta o resultado para o valor de 8 amostras de treino

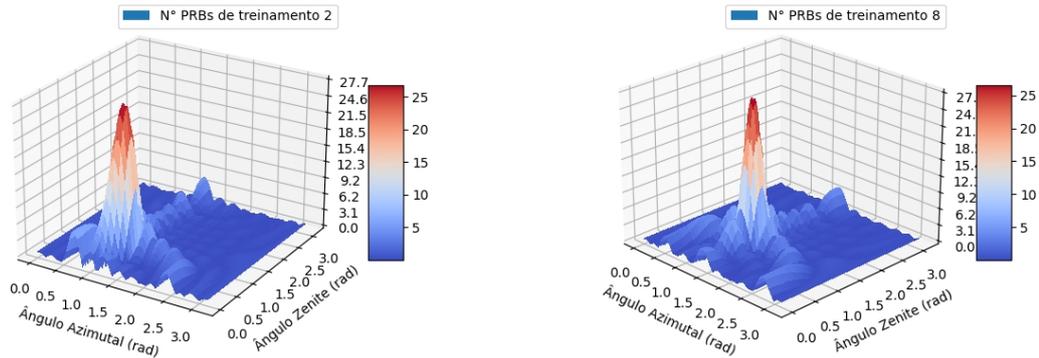


Figura 27 – Resultado para variação no número de amostras de treinamento NLMS, onde: (a) apresenta o resultado para o valor de 2 amostras de treino; (b) apresenta o resultado para o valor de 8 amostras de treino

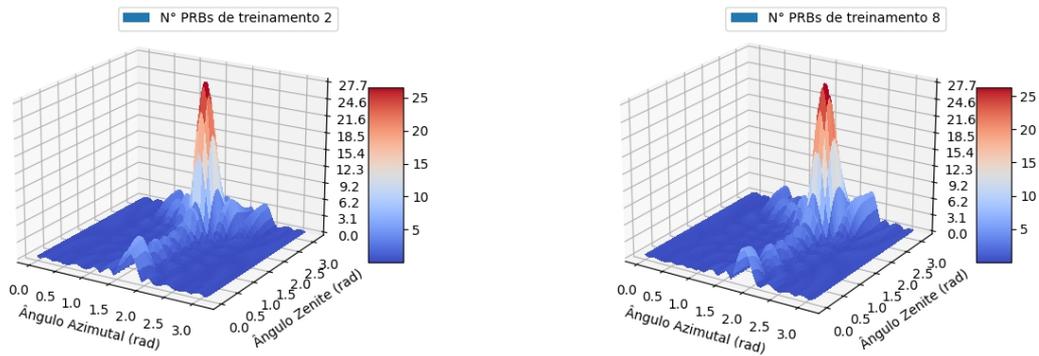


Figura 28 – Resultado para variação no número de amostras de treinamento MRT, onde: (a) apresenta o resultado para o valor de 2 amostras de treino; (b) apresenta o resultado para o valor de 8 amostras de treino

As Figuras 27, 26 e 28 apresentam os resultados para LMS, RLS e MRT. Em todos os casos foi possível formar os feixes e, mesmo com a grande variação no número de PRBs de treinamento, não há impactos visíveis significativos no ganho de *beamforming* e na área do feixe. Isto pois a interpolação realizada é o suficiente para estipular a média dos coeficientes \mathbf{W} para todos os PRBs enviados, outra grande vantagem de se utilizar os algoritmos adaptativos, reservando os demais PRBs para transmissão de dados.

4.2.3 Variação nos desvios em x e y

Os valores máximos de x e y representam o desvio das componentes do sinal em múltiplos percursos.

Para verificar a resistência da técnica de *beamforming* a este efeito, os valores,

inicialmente estabelecidos em 10 para os eixos x e y , foram variados para o valor de 20 em ambos os eixos. Os resultados podem ser vistos em 29, 30 e 31.

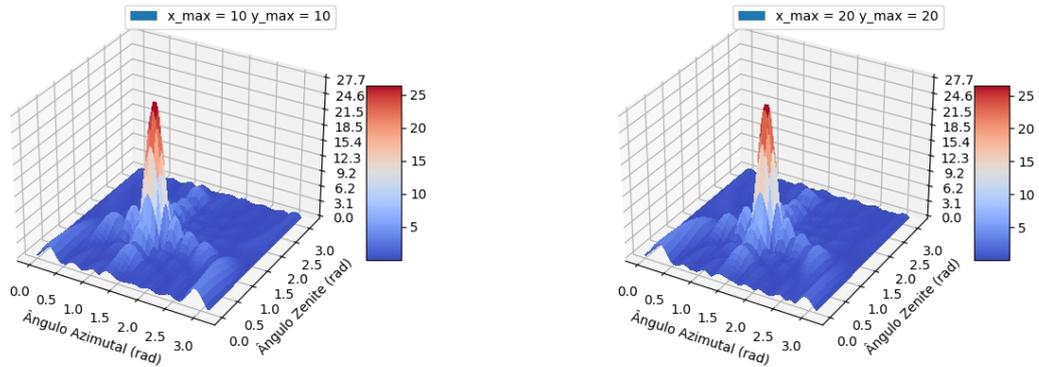


Figura 29 – Resultado para variação no valor de x e y max para o RLS, onde: (a) Apresenta o resultado para x e y igual a 10;(b) Apresenta o resultado para x e y igual a 20

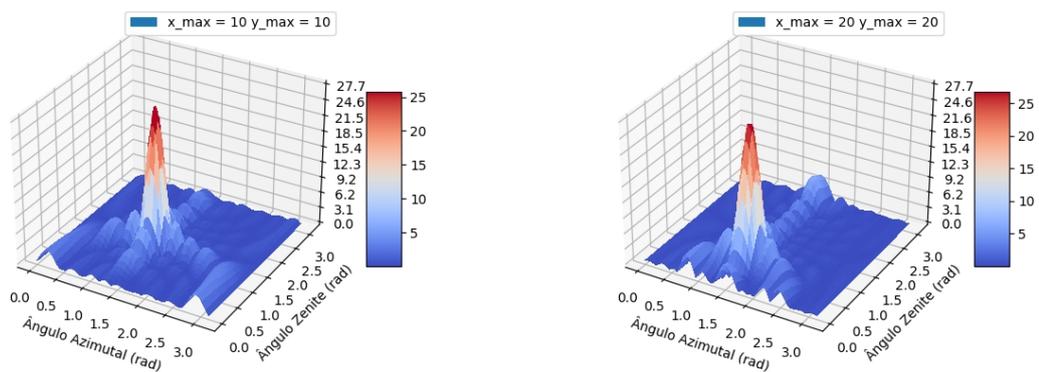


Figura 30 – Resultado para variação no valor de x e y max para o NLMS, onde: (a) Apresenta o resultado para x e y igual a 10;(b) Apresenta o resultado para x e y igual a 20

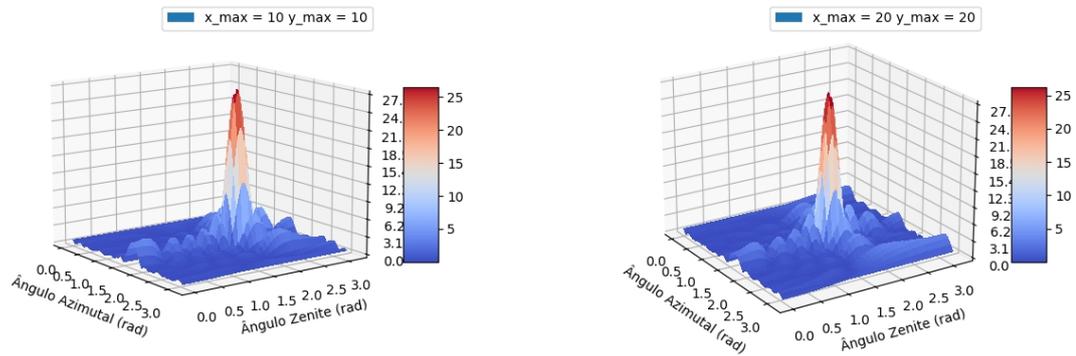


Figura 31 – Resultado para variação no valor de x e y max para o MRT, onde: (a) Apresenta o resultado para x e y igual a 10; (b) Apresenta o resultado para x e y igual a 20

Aqui, verifica-se que para todos os casos, que a variação no ângulo multipercurso, assim como na simulação simplificada, não gera grandes impactos na formação do feixe. Porém, elas ocasionam certo deslocamento da posição central do usuário, que passa a cobrir uma área maior.

4.2.4 Variação no número de usuários

Por fim, foram variados os números de usuários em 2 e 3, verificando o processo de formação de feixe para uma maior quantidade de usuários. Os resultados podem ser vistos abaixo. Os valores de capacidade e capacidade por usuário estão em Mbits/Hz, já o de eficiência espectral em bits/Hz.

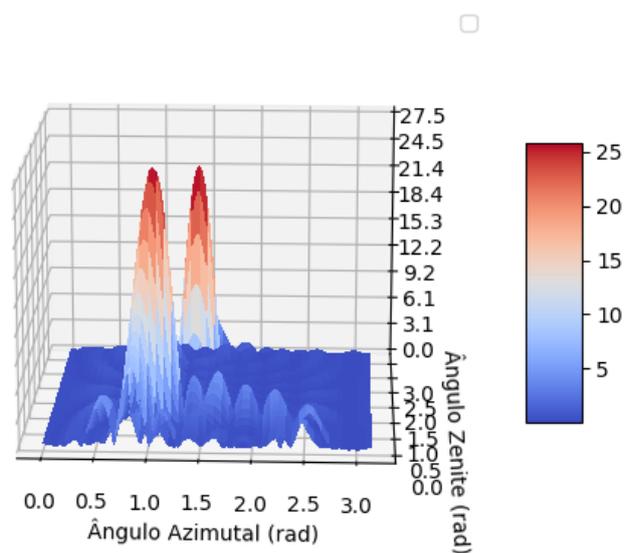


Figura 32 – Resultado para 2 usuários para o RLS

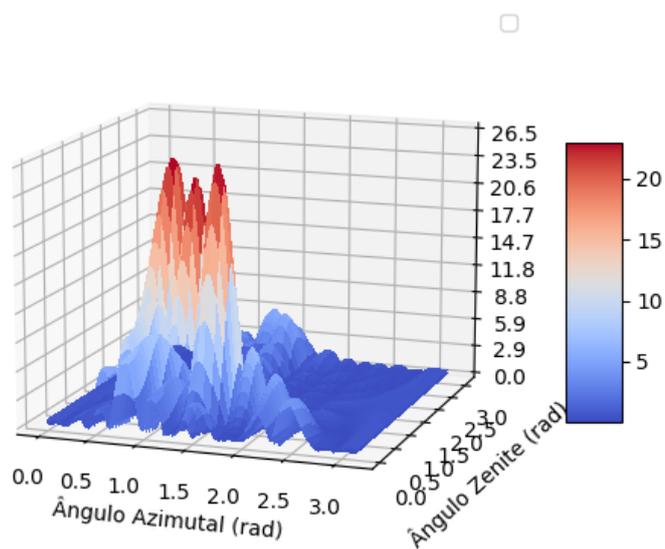


Figura 33 – Resultado para 3 usuários para o RLS

N° usuários	Capacidade	Eficiência espectral	capacidade por usuário
2	3094.65	1.43	2055.58 , 1039.074
3	795.577	0.368	178.41, 40.18, 576.98

Tabela 7 – Valores de capacidade e eficiência espectral RLS para múltiplos usuários

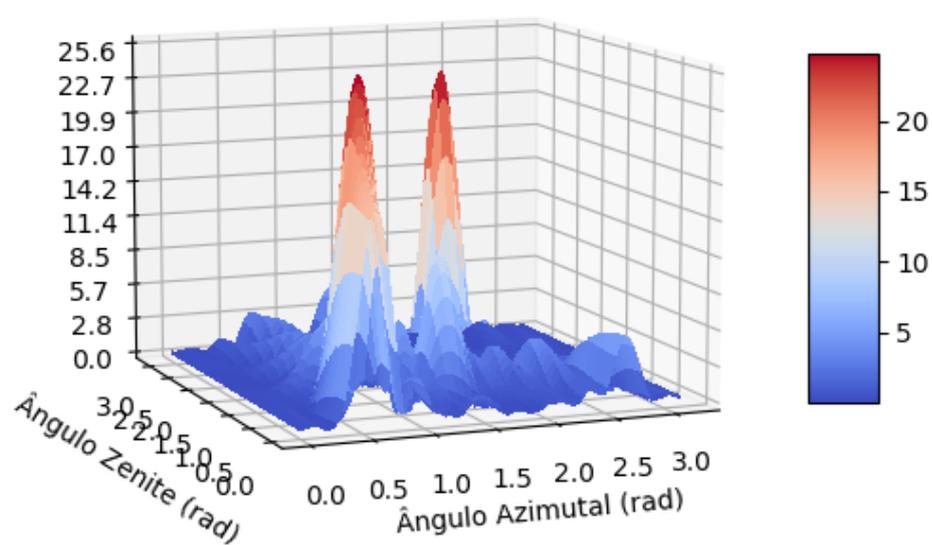


Figura 34 – Resultado para 2 usuários para o NLMS

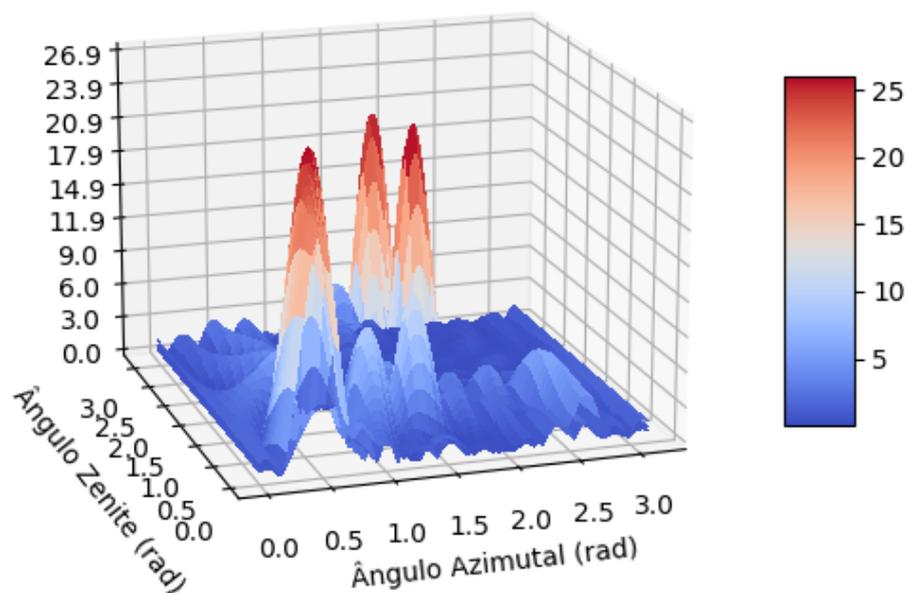


Figura 35 – Resultado para 3 usuários para o NLMS

N° usuários	Capacidade	Eficiência espectral	capacidade por usuário
2	224.25	0.103	86.11, 138.13
3	301.71	0.139	25.19, 8.29, 268.22

Tabela 8 – Valores de capacidade e eficiência espectral NLMS para múltiplos usuários

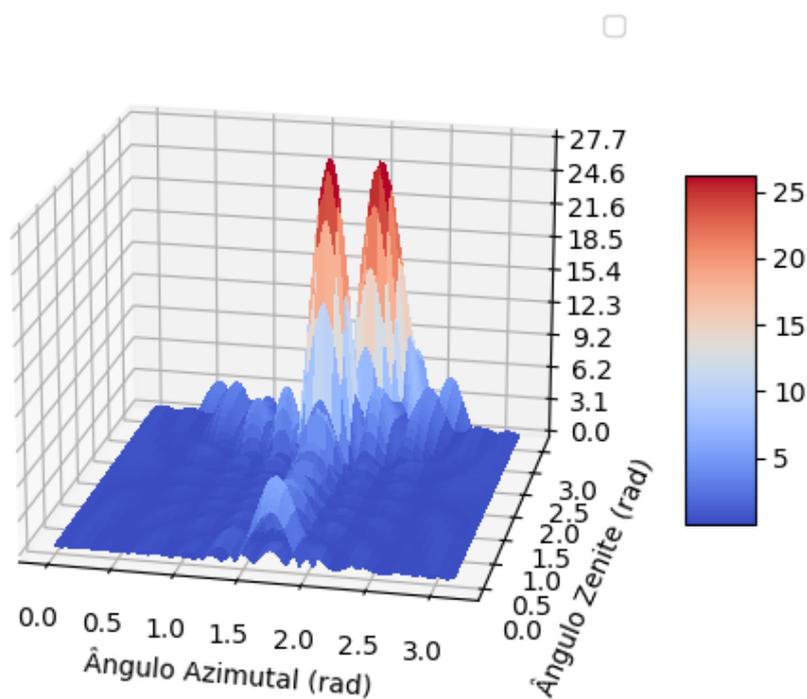


Figura 36 – Resultado para 2 usuários para o MRT

Nº usuários	Capacidade	Eficiência espectral	capacidade por usuário
2	1555.36	0.720	1405.24, 150.11
3	563.82	0.261	274.82, 177.46, 111.53

Tabela 9 – Valores de capacidade e eficiência espectral MRT para múltiplos usuários

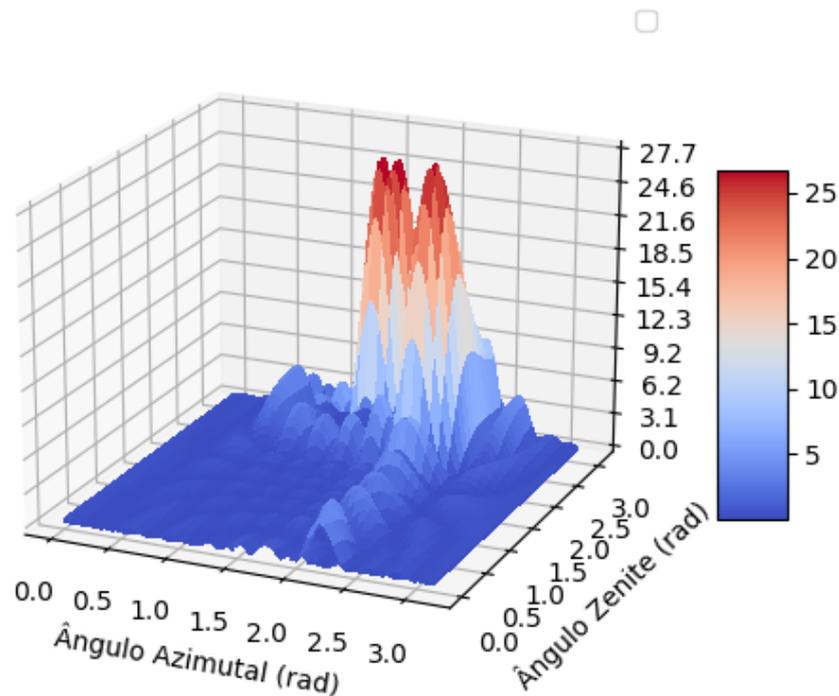


Figura 37 – Resultado para 3 usuários para o MRT

Através das Tabelas 7, 8 e 9, percebe-se que, apesar de formar o feixe na direção de interesse, os algoritmos não realizam uma distribuição equivalente dos recursos do canal, e que os valores de eficiência e capacidade do sistema/usuário tornam-se, de certa forma, aleatórios a depender das características do canal, ao contrário do visto na Tabela 4.

4.3 Conclusões do capítulo

Após as simulações do processo de *beamforming*, conclui-se que os parâmetros que têm o maior impacto no resultado final do feixe são o número de antenas, o valor de SNR e a quantidade de usuários. O número de antenas influencia diretamente no ganho de *beamforming*, e o algoritmo mostra-se eficaz na redução de interferências mesmo em situações de baixo SNR. Quanto à quantidade de usuários, a formação de feixes é possível, mas quanto maior a quantidade, mais complexa a tarefa.

A escolha da sequência de símbolos piloto pode ser feita de maneira ótima, e também influencia na eficácia do processo, sendo que a escolha da menor sequência possível e que forneça a maior SNR é a mais recomendada, para que se gaste a menor quantidade de recursos possível. O azimute influencia na localização do usuário, o que faz com que o

feixe se desloque ao longo do eixo x no sistema de coordenadas cartesianas, enquanto que o fator de Rice não interfere diretamente no resultado de *beamforming*, mas representa características inerentes da distribuição e multipercurso do enlace.

Quanto aos valores de eficiência, capacidade do sistema e capacidade por usuário, percebe-se que, para um usuário, estes valores são estáveis e bem próximos para os algoritmos implementados de NLMS e RLS, já o MRT apresenta uma certa vantagem quando comparado aos dois. Porém, para uma quantidade de usuários maior que um, esses valores passam a ser aleatórios e dependentes tanto das características do canal, quanto da posição e quantidade de usuários.

Foi comprovada, então, a implementação e a eficácia dos algoritmos NLMS e RLS no processo de conformação de feixes, com destaque para o algoritmo NLMS devido à sua baixa complexidade computacional e resultados satisfatórios, e, para o RLS, por fornecer resultados efetivos, apesar da maior complexidade. Além disso, ambos os algoritmos permitem a formação de feixes sem a necessidade de estimar o canal. Este fato possibilita uma formação mais rápida e eficiente, já que a ERB pode receber as amostras de símbolos piloto ao mesmo tempo em que o feixe é formado.

5 Conclusão

Através deste trabalho constata-se que a taxa de transmissão de dados está em crescimento exponencial e que o espectro eletromagnético é um recurso limitado. Desta forma, novas técnicas são desenvolvidas e aplicadas em enlaces de comunicação para que, de diferentes formas, este recurso tão valioso seja melhor utilizado.

Uma destas técnicas, tema de estudo deste trabalho, é a de *beamforming*, oriunda de cenários de comunicação caracterizados como MIMO, aqui formada através de algoritmos adaptativos, em específico, o NLMS e o RLS.

Constata-se também que os simuladores de canais de rádio tem se apresentado como uma grande ferramenta de auxílio no desenvolvimento de pesquisas, principalmente de tecnologias emergentes, devido a sua capacidade cada vez maior de simular cenários realísticos de telecomunicações.

Assim, este trabalho implementou os algoritmos adaptativos para realização da técnica de *beamforming* no simulador de canais de rádios heterogêneo *Hermespy*, através de classes codificadas na linguagem *python*. Também foi feita a avaliação comparativa entre os algoritmos implementados pelo autor deste trabalho e o algoritmo MRT, que já constava como funcionalidade do código, através da simulação de um cenário macro urbano e realizando diversas alterações de parâmetro de entrada a fim de avaliar sua performance, tanto graficamente, quanto através dos valores de eficiência espectral, capacidade total do sistema e capacidade por usuário.

Sob diferentes óticas, os algoritmos provaram-se capazes de realizarem o *beamforming* de maneira ótima e com baixa complexidade computacional. Constatou-se que os fatores que provocam maior alteração no resultado da conformação de feixes são o número de antenas presentes no enlace, o valor da SNR e o número de usuários. Para os demais valores, algumas alterações foram observadas, principalmente no que se refere ao ganho obtido. Para máxima eficácia, existem estudos, citados ao longo deste trabalho, que comprovam que estes valores podem ser escolhidos de maneira ótima, a fim de se obter o melhor ganho possível.

5.1 Futuros trabalhos

Para futuros trabalhos pretende-se abordar os seguintes tópicos:

- Simular outros cenários disponíveis no *Hermespy*.

- Implementar novos algoritmos adaptativos ao simulador e comparar a performance com os aqui implementados.
- Simular usuários em movimento.

Referências

- 3GPP. Technical Specification (TS), *5G;Procedures for the 5G system*. 2018. Version 15.2.0. Disponível em: <https://www.etsi.org/deliver/etsi_ts/123500_123599/123502/15.02.00_60/ts_123502v150200p.pdf>. Citado na página 17.
- 3GPP. Technical report (TR), *Study on 3D channel model for LTE*. 2018. Version 12.7.0. Disponível em: <<https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2574>>. Citado na página 29.
- ABDI, A. et al. On the estimation of the k parameter for the rice fading distribution. *Communications Letters, IEEE*, v. 5, p. 92 – 94, 04 2001. Citado na página 49.
- ADLER, J.; KRONAUER, T.; BARRETO, A. N. Hermespy: An open-source link-level evaluator for 6g. *IEEE Access*, v. 10, p. 120256–120273, 2022. Citado na página 18.
- AGIWAL, M.; ROY, A.; SAXENA, N. Next generation 5g wireless networks: A comprehensive survey. *IEEE Communications Surveys Tutorials*, v. 18, n. 3, p. 1617–1655, 2016. Citado na página 17.
- BJÖRNSON, E.; HOYDIS, J.; SANGUINETTI, L. Massive MIMO networks: Spectral, energy, and hardware efficiency. *Foundations and Trends® in Signal Processing*, v. 11, n. 3-4, p. 154–655, 2017. ISSN 1932-8346. Disponível em: <<http://dx.doi.org/10.1561/20000000093>>. Citado 3 vezes nas páginas 9, 21 e 27.
- CHAITANYA, M.; REDDY, I. S. P. System performance of mimo mrt/mrc system under perfect csi. p. 49, 02 2020. Citado na página 38.
- DINIZ, P. Adaptive filtering: Algorithms and practical implementation. In: _____. [S.l.: s.n.], 2008. Citado 5 vezes nas páginas 31, 32, 33, 34 e 36.
- ERICSSON. *Mobile data traffic outlook*. 2021. Disponível em: <<https://www.ericsson.com/en/mobility-report/dataforecasts/mobile-trafficforecast>>. Citado 3 vezes nas páginas 9, 18 e 19.
- GKONIS, P.; PANAGIOTIS, T.; KAKLAMANI, D. A comprehensive study on simulation techniques for 5g networks: State of the art results, analysis, and future challenges. *Electronics*, v. 9, p. 468, 03 2020. Citado na página 18.
- HUAWEI. *Huawei AAU Introduction*. 2022. Disponível em: <<https://forum.huawei.com/enterprise/en/huawei-aau-introduction/thread/827987-100305>>. Citado na página 50.
- LARSSON, E. G. et al. Massive mimo for next generation wireless systems. *IEEE Communications Magazine*, v. 52, n. 2, p. 186–195, 2014. Citado na página 23.
- MARZETTA, T. L. Noncooperative cellular wireless with unlimited numbers of base station antennas. *IEEE Transactions on Wireless Communications*, v. 9, n. 11, p. 3590–3600, 2010. Citado na página 23.

- MARZETTA, T. L. Massive mimo: An introduction. *Bell Labs Technical Journal*, v. 20, p. 11–22, 2015. Citado 3 vezes nas páginas 17, 21 e 52.
- MEENA, K.; KABILAN, A. A simple adaptive beamforming algorithm with interference suppression. *International Journal of Engineering and Technology*, v. 1, 01 2009. Citado 2 vezes nas páginas 9 e 26.
- MENDES, L. L. Modelos matemáticos para estimação do desempenho de sistemas de multiplexação por divisão em frequências ortogonais. *Universidade Estadual de Campinas*, p. 173, 2007. Citado 2 vezes nas páginas 23 e 24.
- MOON, T. K.; STIRLING, W. C. Mathematical methods and algorithms for signal processing. *Pearson Prentice Hall*, 2000. Citado na página 35.
- NAHUZ, C. S. Algoritmo adaptativo do tipo lms com soma do erro. *Universidade Federal do Maranhão*, 2016. Citado na página 34.
- NETO, M. U.; GOMES, L. d. C. T. Filtragem adaptativa de sinais. *Intellectus*, v. 6, n. 8, 2010. Citado 2 vezes nas páginas 9 e 33.
- NOH, S. et al. Pilot beam pattern design for channel estimation in massive mimo systems. *IEEE Journal of Selected Topics in Signal Processing*, v. 8, n. 5, p. 787–801, 2014. Citado na página 47.
- RODRIGUES, M. E. da C. *ASPECTOS DE RÁDIO - PROPAGAÇÃO*. 2003. Disponível em: <http://www.wirelessbrasil.org/wirelessbr/colaboradores/marcio_rodrigues/propagacao/prop_04.html>. Citado 2 vezes nas páginas 28 e 49.
- SANTOS, D. M. S. d. Seleção de símbolos piloto em sistemas de comunicação sem fio. *Universidade Federal do Ceará*, p. 66, 2016. Citado 3 vezes nas páginas 9, 21 e 25.
- SKONDRAS, E. et al. A downlink scheduler supporting real time services in lte cellular networks. In: . [S.l.: s.n.], 2015. p. 1–6. Citado na página 24.
- SO, J. et al. Pilot signal design for massive mimo systems: A received signal-to-noise-ratio-based approach. *IEEE Signal Processing Letters*, v. 22, n. 5, p. 549–553, 2015. Citado na página 47.
- SOHRABI, F.; YU, W. Hybrid digital and analog beamforming design for large-scale antenna arrays. *IEEE Journal of Selected Topics in Signal Processing*, v. 10, n. 3, p. 501–513, 2016. Citado na página 18.
- VEEN, B. V.; BUCKLEY, K. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine*, v. 5, n. 2, p. 4–24, 1988. Citado na página 25.
- VERDU, S. Spectral efficiency in the wideband regime. *IEEE Transactions on Information Theory*, v. 48, n. 6, p. 1319–1343, 2002. Citado na página 30.
- VISWANATHAN, M. *Model and characterize MIMO channels*. 2014. Disponível em: <<https://www.gaussianwaves.com/2014/08/characterizing-a-mimo-channel/>>. Citado 2 vezes nas páginas 9 e 22.

WOODBURY, M.; STATISTICS, P. U. D. of. *Inverting Modified Matrices*. Department of Statistics, Princeton University, 1950. (Memorandum Report / Statistical Research Group, Princeton). Disponível em: <https://books.google.com.br/books?id=_zAnzgEACAAJ>. Citado na página 37.