



**COPPE/UFRJ**

NOVOS ALGORITMOS DE ESTIMAÇÃO APLICADOS A SISTEMAS DE  
COMUNICAÇÕES

Tadeu Nagashima Ferreira

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientadores: Paulo Sergio Ramirez Diniz  
Sergio Lima Netto

Rio de Janeiro  
Novembro de 2009

NOVOS ALGORITMOS DE ESTIMAÇÃO APLICADOS A SISTEMAS DE  
COMUNICAÇÕES

Tadeu Nagashima Ferreira

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ  
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE)  
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS  
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR  
EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

---

Prof. Paulo Sergio Ramirez Diniz, Ph.D.

---

Prof. Sergio Lima Netto, Ph.D.

---

Prof. Marcello Luiz Rodrigues de Campos, Ph.D.

---

Prof. José Antonio Apolinário Jr., D.Sc.

---

Prof. Raimundo Sampaio Neto, Ph.D.

---

Prof. Renato da Rocha Lopes, Ph.D.

RIO DE JANEIRO, RJ – BRASIL

NOVEMBRO DE 2009

Ferreira, Tadeu Nagashima

Novos Algoritmos de Estimação Aplicados a Sistemas de Comunicações/Tadeu Nagashima Ferreira. – Rio de Janeiro: UFRJ/COPPE, 2009.

XVII, 137 p.: il.; 29, 7cm.

Orientadores: Paulo Sergio Ramirez Diniz

Sergio Lima Netto

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2009.

Referências Bibliográficas: p. 132 – 137.

1. estimação. 2. processamento de sinais. 3. array de antenas. 4. OFDM. I. Diniz, Paulo Sergio Ramirez *et al.*. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

# Agradecimentos

A meus orientadores, Prof. Paulo Diniz e Prof. Sergio Lima Netto. À minha namorada, Luciana Fernandes. Ao CNPq e à Faperj. A meus pais, Nair e Helio. A meus grandes amigos (quase familiares), Ana Luísa Santos e Leonardo Baltar, este também um parceiro no desenvolvimento de um dos algoritmos. Aos meus colegas de turma de doutorado, Michel Tcheou, Filipe Diniz, Maurício Quélhas e Rodrigo Torres. Aos professores Luiz Wagner Biscainho, Marcello Campos, Raimundo Sampaio, José Antonio Apolinário, Renato Lopes, Lisandro Lovisolo, Amaro Lima, Fábio Freland e Eduardo Nunes. Aos amigos e colegas de trincheira Alan Tygel, Leonardo Nunes, Wallace Martins, Markus Lima, Thiago Prego, Fabiano Castoldi, Alessandro Dutra, André Targino, Guilherme Pinto, Camila Gussen, Ana Fernanda Quaresma, Adriana Schultz, Paulo Cezar Martins, Rafael RAFS, João Dias, Rafael de Barretos, Axel Hollanda, Bruno Bispo, Alexandre Leizor, Rafael de Jesus, Mamour Ndiaye, Alexandre Ciancio, Felipe Grael, Waldir Sabino, Flávio Ávila e Rafael Paiva. Aos que esqueci, por distração. À amiga Vilma Rangel. Aos amigos do Goethe, Renato Borges, Kristoffer Hallam, Glauco Gavioli, Marlan Marinho, Patricia Burrowes, Fernanda Fonseca, Sophia Otero, Paulo Cuadrat, Fernandão, Joana Foutora e Leonardo Montanholi. Aos amigos de outras épocas, Édson Hiroshi, Roberto Hori, Eduardo Szrajbman, Carlo Marcello Siqueira, Marcello Artimos, Luis Guilherme Uzeda, Nuno Rodrigues, Régis Yamashita, Nathan Feldman, Jürgen Wittmann, Rodrigo Meirelles, Aurélio Magalhães e Daniel Bonolo. Aos funcionários do LPS, Michelle Nogueira, Amanda Loiola e Waldyr Leite. À funcionária do Burguesão, Maria “do Café”. Vielen Dank, Professor Wolfgang Utschik, unseres Gespräch war sehr wichtig. Ao professor de ensino médio Gerson Costamilan, a primeira pessoa que me explicou o que é um doutorado. À cafeína e à cevada, Thanathos e Eros no sangue.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

## NOVOS ALGORITMOS DE ESTIMAÇÃO APLICADOS A SISTEMAS DE COMUNICAÇÕES

Tadeu Nagashima Ferreira

Novembro/2009

Orientadores: Paulo Sergio Ramirez Diniz

Sergio Lima Netto

Programa: Engenharia Elétrica

A estimação da direção de chegada (DoA - *direction-of-arrival*) de uma frente de onda em um receptor permite o ajuste de coeficientes de filtros espaciais em um sistema com um conjunto (*array*) de antenas de recepção. Isso permite uma separação espacial mais precisa das fontes, o que aumenta a capacidade de um sistema de comunicações.

A estimação do desvio de frequência de portadoras (CFO - *carrier frequency offset*) em sistemas OFDM (*orthogonal frequency division multiplexing*) permite que tal distorção seja compensada, o que restaura a ortogonalidade das portadoras, facilitando a eliminação de ISI (*intersymbol interference*).

Este trabalho apresenta um conjunto de algoritmos paramétricos de baixa complexidade com bom desempenho em termos de taxa de erros quando aplicados aos contextos de estimação de DoA para sistemas com múltiplas antenas e de CFO para sistemas OFDM. A diminuição da complexidade é um dos fatores que viabiliza, em ambos os casos, a implementação dos algoritmos em sistemas comerciais. A redução da complexidade computacional ocorre, na maioria dos algoritmos, sem aumento na taxa de erro dos algoritmos propostos em comparação com algoritmos da literatura.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

NEW ESTIMATION ALGORITHMS APPLIED TO COMMUNICATION  
SYSTEMS

Tadeu Nagashima Ferreira

November/2009

Advisors: Paulo Sergio Ramirez Diniz

Sergio Lima Netto

Department: Electrical Engineering

The Direction-of-Arrival (DoA) estimation for a mixture of transmitted signals in a receiver is a useful procedure for designing the coefficients of spatial filters in a system with a multiple-antenna receiver. Spatial filtering is necessary for an accurate source separation, which increases the overall capacity of a communication system.

Carrier frequency offset (CFO) estimation in an OFDM (orthogonal frequency division multiplexing) system is the first step for the mitigation of CFO distortion. Frequency offset is responsible for generating non-orthogonality among transmitting carriers, which is particularly harmful for the process of ISI (intersymbol interference) elimination.

This thesis presents a set of low-complexity parametric algorithms with good performance for both scenarios of DoA estimation in multi-antenna systems as well as of CFO estimation in OFDM systems. Low complexity is one of the characteristics which makes both estimation systems more feasible for implementation in commercial systems. For the majority of the new algorithms, the reduction in computational complexity does not increase the error rate of the proposed algorithms, when compared to similar ones in the literature.

# Sumário

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Tabelas</b>	<b>xiii</b>
<b>Lista de Abreviaturas</b>	<b>xv</b>
<b>1 Introdução da Tese</b>	<b>1</b>
1.1 Considerações Iniciais . . . . .	1
1.2 Motivações . . . . .	1
1.3 Objetivos . . . . .	3
1.4 Sobre as Variáveis Utilizadas . . . . .	5
1.5 Organização do Trabalho . . . . .	6
<b>2 Algoritmos Existentes para Equalização e Estimação de DoA</b>	<b>7</b>
2.1 Introdução do Capítulo . . . . .	7
2.2 Equalização Cega . . . . .	9
2.2.1 Descrição do Ambiente . . . . .	9
2.2.2 Algoritmo TXK para Equalização Cega . . . . .	12
2.3 Estimação de DoA . . . . .	16
2.3.1 Métodos Paramétricos . . . . .	17
2.3.2 Descrição do Ambiente . . . . .	21
2.3.3 Algoritmo ESPRIT para Estimação de DoA . . . . .	24
2.3.4 Processamento no <i>Beamspace</i> DFT . . . . .	27
2.3.5 <i>Unitary</i> ESPRIT . . . . .	32
2.4 Métodos do tipo MP ( <i>Matrix Pencil</i> ) . . . . .	40
2.5 Discussões do Capítulo . . . . .	43

2.6	Conclusões do Capítulo . . . . .	47
<b>3</b>	<b>Novos Algoritmos para Estimação de DoA</b>	<b>49</b>
3.1	Introdução do Capítulo . . . . .	49
3.2	Estendendo o TXK para a estimação de DoA . . . . .	50
3.3	<i>Covariance-Based DoA Estimation</i> (CB-DoA) . . . . .	53
3.3.1	Comparações entre CB-DoA e TLS-ESPRIT . . . . .	54
3.3.2	CB-DoA e <i>Cramer-Rao Lower Bound</i> . . . . .	59
3.4	<i>Beamspace</i> CB-DoA . . . . .	60
3.4.1	Comparações entre <i>Beamspace</i> CB-DoA e <i>Beamspace</i> ESPRIT	63
3.4.2	Comparações com o CRLB . . . . .	66
3.4.3	Comparações entre <i>Beamspace</i> CB-DoA e <i>Elementspace</i> CB- DoA . . . . .	67
3.5	<i>Real</i> CB-DoA . . . . .	68
3.5.1	Comparações entre <i>Real</i> CB-DoA e <i>Unitary</i> ESPRIT . . . . .	71
3.6	<i>Direct-Data</i> DoA (DD-DoA) . . . . .	74
3.6.1	Comparações entre DD-DoA e TLS-ESPRIT . . . . .	77
3.6.2	Comparações entre DD-DoA e CRLB . . . . .	80
3.7	Discussões do Capítulo . . . . .	81
3.7.1	Sobre as Comparações com o TLS-ESPRIT . . . . .	81
3.7.2	Comparações com outros Algoritmos . . . . .	82
3.7.3	Sobre Questões Internas dos Algoritmos Propostos . . . . .	84
3.8	Conclusões do Capítulo . . . . .	86
<b>4</b>	<b>Algoritmos para Estimação de CFO</b>	<b>87</b>
4.1	Introdução do Capítulo . . . . .	87
4.2	CFO em Sistemas OFDM . . . . .	87
4.2.1	Descrição do Ambiente . . . . .	88
4.2.2	Algoritmo ESPRIT para Estimação de CFO para OFDM . . . . .	91
4.3	Estendendo o TXK para Estimação de CFO . . . . .	96
4.4	<i>Covariance-Based CFO Estimation</i> (CB-CFO) . . . . .	98
4.4.1	Comparações entre CB-CFO e TLS-ESPRIT . . . . .	100
4.5	<i>Unitary</i> ESPRIT para Estimação de CFO . . . . .	103



4.5.1	Comparações entre ESPRIT e <i>Unitary</i> ESPRIT para Es- timação de CFO . . . . .	107
4.6	Discussões do Capítulo . . . . .	111
4.7	Conclusões do Capítulo . . . . .	118
<b>5</b>	<b>Conclusões da Tese</b>	<b>120</b>
5.1	Sobre o Documento . . . . .	120
5.2	Métodos Desenvolvidos . . . . .	121
5.3	Propostas de Trabalhos Futuros . . . . .	123
<b>A</b>	<b>Análise de MSE</b>	<b>126</b>
	<b>Publicações do Autor</b>	<b>131</b>
	<b>Referências Bibliográficas</b>	<b>132</b>

# Lista de Figuras

1.1	Esquema com os algoritmos apresentados na tese. . . . .	5
2.1	Ambiente de comunicações associado ao processo de equalização cega. . . . .	10
2.2	Diagrama de blocos representando o processo de estimação do algoritmo TXK. . . . .	16
2.3	Etapas dos métodos SSI. . . . .	19
2.4	Ambiente de múltiplas fontes de transmissão e múltiplas antenas de recepção. . . . .	22
2.5	Ângulo $\theta$ medido por algoritmos para estimação de DoA. . . . .	23
2.6	Diagrama de blocos representando o processo de estimação do algoritmo ESPRIT. . . . .	27
2.7	Interpretação da projeção no <i>Beamspace</i> DFT vista como varredura num espaço de ângulos. . . . .	28
2.8	Escolhas de <i>sub-arrays</i> de recepção no <i>Elementspace</i> ESPRIT e no <i>Beamspace</i> ESPRIT. No caso <i>elementspace</i> , antenas $(i, i + 1)$ , $i$ ímpar formam <i>doublets</i> com um <i>sub-array</i> de antenas com padrão cinza e outro representado com um padrão quadriculado. No caso <i>beamspace</i> , as antenas com padrão cinza quadriculado pertencem a ambos <i>sub-arrays</i> , com uma sobreposição de elementos entre os <i>doublets</i> . . . . .	29
2.9	Antenas de recepção com as restrições geométricas referentes à invariância à rotação e as novas restrições para <i>arrays</i> centro-hermitianos. Antenas representadas com o mesmo padrão de preenchimento possuem as mesmas características de recepção. . . . .	33
2.10	Representação dos algoritmos apresentados no Capítulo 2, compreendendo os ambientes de equalização cega e estimação de DoA. . . . .	44

3.1	Ambiente de identificação que engloba, como casos particulares, os ambientes de equalização cega e de estimação de DoA. . . . .	50
3.2	Diagrama de blocos representando o processo de estimação do algoritmo CB-DoA. . . . .	55
3.3	MSE para TLS-ESPRIT e CB-DoA em função da SNR para $M = 4$ antenas de transmissão e $N = 9$ <i>doublets</i> de recepção. . . . .	57
3.4	MSE para TLS-ESPRIT e CB-DoA em função da SNR para $M = 7$ antenas de transmissão e $N = 12$ <i>doublets</i> de recepção. . . . .	58
3.5	MSE para CB-DoA e TLS-ESPRIT, e o CRLB aproximado, com $M = 1$ e $N = 4$ . . . . .	60
3.6	Esquema de funcionamento para o algoritmo <i>BeamSpace</i> CB-DoA. . .	61
3.7	Diagrama de blocos representando o processo de estimação do algoritmo <i>BeamSpace</i> CB-DoA. . . . .	63
3.8	MSE medido para os algoritmos <i>BeamSpace</i> ESPRIT e <i>BeamSpace</i> CB-DoA, usando 2 fontes e 18 antenas de recepção, em função do número de <i>beams</i> . . . . .	66
3.9	MSE para os algoritmos <i>BeamSpace</i> ESPRIT e <i>BeamSpace</i> CB-DoA, para $M = 2$ fontes e $N = 18$ antenas de recepção, em função da SNR, para 12 <i>beams</i> . CRLB também está incluído. O uso de 18 antenas neste cenário corresponde às versões <i>elementspace</i> de cada algoritmo. . . . .	67
3.10	Diagrama de blocos representando o processo de estimação do algoritmo <i>Real</i> CB-DoA. . . . .	71
3.11	Comparação em termos de MSE para os algoritmos <i>Unitary</i> ESPRIT e <i>Real</i> CB-DoA em dois cenários de simulações, utilizando 2 e 4 fontes e um <i>array</i> de recepção de 12 antenas, em ambos os cenários. . . . .	75
3.12	Diagrama de blocos representando o processo de estimação do algoritmo DD-DoA. . . . .	77

3.13	Comparação entre TLS-ESPRIT, na versão que opera diretamente sobre os dados, e DD-DoA em termos de MSE para $M = 2$ fontes and $P = 14$ antenas, e também para o DD-DoA com $M = 2$ fontes e $P = 26$ antenas. A curva para MSE teórico, calculado utilizando a equação (A.14), para $M = 2$ fontes e $P = 14$ antenas também pode ser vista. . . . .	80
3.14	MSE para o algoritmo DD-DoA para $M = 1$ fonte e $P = 8$ antenas, e seu limite teórico CRLB. . . . .	81
4.1	Sistema com multiportadoras, do tipo OFDM. O CFO é modelado como um ruído multiplicativo que interfere com o sinal na saída do canal. . . . .	88
4.2	Ambiente de identificação que engloba, como casos particulares, os ambientes de equalização cega, de estimação de DoA e de estimação de CFO em OFDM. . . . .	96
4.3	Diagrama de blocos representando o processo de estimação do algoritmo CB-CFO. . . . .	100
4.4	NMSE para ESPRIT e CB-CFO. . . . .	103
4.5	Diagrama de blocos representando o processo de estimação do algoritmo <i>Unitary</i> ESPRIT. . . . .	106
4.6	NMSE para ESPRIT e <i>Unitary</i> ESPRIT para canais rural (R) e urbano (U) das especificações do LTE [5]. . . . .	110
4.7	Representação dos algoritmos apresentados nos Capítulos 2 e 4, compreendendo os ambientes de equalização cega, estimação de DoA e estimação de CFO para OFDM. . . . .	111
4.8	<i>Array</i> de antenas de recepção. . . . .	112
4.9	Representação do algoritmo GTXK, usando um diagrama de blocos. .	116

# Lista de Tabelas

2.1	Resumo das operações do algoritmo TXK. . . . .	16
3.1	Equivalências entre TXK estendido e o ambiente de estimação de DoA.	53
3.2	Comparação entre os algoritmos TLS-ESPRIT e CB-DoA. . . . .	56
3.3	Comparação para as operações requeridas por TLS-ESPRIT e CB-DoA.	57
3.4	Tempo médio de execução para TLS-ESPRIT and CB-DoA para número distintos de antenas transmissoras ( $M$ ) e <i>doublets</i> de recepção ( $N$ ). . . . .	59
3.5	Resumo comparativo dos algoritmos <i>Beamspace</i> ESPRIT e <i>Beamspace</i> CB-DoA. . . . .	64
3.6	Comparação para as operações requeridas por <i>beamspace</i> ESPRIT e <i>beamspace</i> CB-DoA. . . . .	65
3.7	Descrição resumida dos algoritmos <i>Unitary</i> ESPRIT e <i>Real</i> CB-DoA.	72
3.8	Comparação dos algoritmos <i>Unitary</i> ESPRIT e <i>Real</i> CB-DoA em termos do número de operações. . . . .	74
3.9	Descrição resumida dos algoritmos DD-DoA e TLS-ESPRIT, na versão que opera diretamente sobre os dados. . . . .	78
3.10	Comparação entre TLS-ESPRIT, em sua versão que opera diretamente sobre os dados, e DD-DoA em termos de número de operações com matrizes. . . . .	79
3.11	Comparação para as operações requeridas por LS-ESPRIT e CB-DoA.	82
3.12	Comparação da quantidade de flops necessários em operações sobre dados reais e complexos, considerando-se a quantidade de operações reais. . . . .	85
4.1	Tabela resumindo parte dos efeitos do CFO sobre um sistema OFDM.	91

4.2	Equivalências entre TXK estendido e o ambiente de estimação de CFO.	99
4.3	Comparação dos algoritmos TLS-ESPRIT [6] e CB-CFO.	101
4.4	Comparação para as operações requeridas por TLS-ESPRIT e CB-CFO.	102
4.5	Comparação dos algoritmos TLS-ESPRIT [6] e <i>Unitary</i> ESPRIT para estimação de CFO.	108
4.6	Comparação entre o número de operações dos algoritmos ESPRIT e <i>Unitary</i> ESPRIT para estimação de CFO.	109
4.7	Nova identificação de entidades do GTXK com TXK, CB-DoA (versões <i>elementspace</i> e <i>beamspace</i> ), e CB-CFO.	117

# Lista de Abreviaturas

3GPP	<i>3<sup>rd</sup> Generation Partnership Project</i> , p. 46
AIC	<i>Akaike's Information Theoretic Criteria</i> , p. 18
AWGN	<i>Additive White Gaussian Noise</i> , p. 9
BER	<i>Bit Error Rate</i> , p. 46
CB-CFO	<i>Covariance-Based Carrier Frequency Offset Estimation</i> , p. 4
CB-DoA	<i>Covariance-Based Direction-of-Arrival Estimation</i> , p. 4
CCI	<i>Co-channel interference</i> , p. 2
CDMA	<i>Code Division Multiple Access</i> , p. 3
CFO	<i>Carrier Frequency Offset</i> , p. 3
CH	<i>Centro-Hermitiano</i> , p. 33
CMA	<i>Constant-Modulus Algorithm</i> , p. 8
CRLB	<i>Cramer-Rao Lower Bound</i> , p. 59
DD-DoA	<i>Direct Data Direction-of-Arrival Estimation</i> , p. 49
DECT	<i>Digital Enhanced Cordless Communications</i> , p. 46
DFT	<i>Discrete Fourier Transform</i> , p. 3
DVB-T	<i>Digital Video Broadcasting - Terrestrial</i> , p. 118
DVB	<i>Digital Video Broadcasting</i> , p. 3
DoA	<i>Direction-of-Arrival</i> , p. 4

ESPRIT	<i>Estimation of Parameters via Rotational Invariance Techniques</i> , p. 4
EVD	<i>Eigenvalue/Eigenvector Decomposition</i> , p. 13
FB	<i>Forward-Backward</i> , p. 36
GEVD	<i>Generalized Eigenvalue/eigenvector Decomposition</i> , p. 19
IBI	<i>Interblock Interference</i> , p. 90
ICI	<i>Intercarrier Interference</i> , p. 90
IDFT	<i>Inverse Discrete Fourier Transform</i> , p. 88
IMT-2000	<i>International Mobile Telecommunications 2000</i> , p. 46
IS-136	<i>Interim Standard 136</i> , p. 46
IS-95	<i>Interim Standard 95</i> , p. 46
ISI	<i>Intersymbol Interference</i> , p. 1
LS	<i>Least-Squares</i> , p. 18
LTE	<i>Long-Term Evolution</i> , p. 2
MAI	<i>Multiple Access Interference</i> , p. 1
MAP	<i>Maximum a Posteriori</i> , p. 1
ML	<i>Maximum Likelihood</i> , p. 1
MP	<i>Matrix Pencil</i> , p. 9
MSE	<i>Mean Square Error</i> , p. 43
MUSIC	<i>Multiple Signal Classification</i> , p. 8
NMSE	<i>Normalized Mean Square Error</i> , p. 102
OFDM	<i>Orthogonal Frequency Division Multiplexing</i> , p. 3
PASTd	<i>Projection Approximation Subspace Tracking</i> , p. 18



SDMA	<i>Space Division Multiple Access</i> , p. 2
SLS	<i>Structured Least-Squares</i> , p. 124
SNR	<i>Signal-To-Noise Ratio</i> , p. 56
SSI	<i>Single Shift-Invariance</i> , p. 17
SVD	<i>Singular Value Decomposition</i> , p. 14
TB	Transformação Bilateral, p. 15
TDMA	<i>Time Division Multiple Access</i> , p. 45
TLS	<i>Total Least-Squares</i> , p. 18
TSR	<i>Translational Series Representation</i> , p. 10
TSUNAMI	<i>Technology in Smart Antennas for the Universal Advanced Mobile Infrastructure</i> , p. 47
TXK	Tong, Xu, Kailath, p. 5
ULA	<i>Uniformly Linear Array</i> , p. 27
WCDMA	<i>Wideband Code Division Multiple Access</i> , p. 47
WiMAX	<i>Wireless Interoperability for Microwave Access</i> , p. 88
ZF	<i>Zero Forcing</i> , p. 15

# Capítulo 1

## Introdução da Tese

### 1.1 Considerações Iniciais

Neste capítulo são apresentados os principais objetivos e motivações para o desenvolvimento deste trabalho, bem como o fluxo de trabalho da tese, a organização geral deste documento e algumas observações sobre a notação utilizada.

### 1.2 Motivações

Aplicações de técnicas de estimação em sistemas de comunicações têm origem em interpretações estatísticas da transmissão da informação. O uso de receptores com base em estimativas do tipo *maximum likelihood* (ML) ou *maximum a posteriori* (MAP) são técnicas consolidadas há aproximadamente meio século [7].

A sofisticação crescente por que passaram os sistemas de comunicações nos últimos anos tem levado ao uso de técnicas cada vez mais avançadas para a recepção de sinais. A necessidade de aumento de banda para novas aplicações tem gerado um aumento de certos tipos de interferência nos sistemas de comunicação, sobretudo a interferência intersimbólica (ISI, *intersymbol interference*) e, marginalmente, a interferência de múltiplo acesso (MAI, *multiple access interference*). A ISI corresponde à interferência entre símbolos transmitidos por uma mesma fonte, causada pela passagem do sinal por um canal que possui banda menor que a do sinal transmitido. Por sua vez, a MAI corresponde à interferência causada por sinais de usuários diferentes que acessam a um mesmo meio de comunicação. Em

contrapartida, foram desenvolvidas novas técnicas de combate a essas interferências, incluindo o uso de diversidade espacial e o uso de múltiplas portadoras no sistema de comunicação.

Com o crescente aumento da banda utilizada na transmissão da informação, os requisitos computacionais para aplicações em tempo real têm se tornado progressivamente mais rígidos. O desenvolvimento de sistemas com uma menor complexidade computacional tem se revelado cada vez mais necessário. Além disso, com o aumento de demanda por tecnologias móveis, os requisitos de complexidade computacional, em especial no *downlink* para o receptor, proporcionaram uma maior integração no nível de circuitos para o dispositivo móvel e uma maior duração da bateria.

Os sistemas de comunicação *broadcasting* por rádio em sua origem utilizaram antenas altas, com grande potência, a fim de atender a uma maior quantidade de usuários. Por outro lado, sistemas de telefonia celular têm como base a divisão espacial do sistema em células, com estações-base associadas a cada uma delas [8]. Esse novo paradigma de projeto, no entanto, gerou o aumento de um outro tipo de interferência, denominada interferência co-canal (CCI, *co-channel interference*), que se caracteriza pela corrupção da informação transmitida, num dado canal, pelo usuário desejado causada pela transmissão de outros usuários, localizados em outras células, quando estão usando esse mesmo canal.

A utilização de múltiplas antenas para transmissão e recepção num sistema de comunicações possibilitou o desenvolvimento de algumas novas técnicas para o combate à CCI baseadas no aproveitamento de diversidade espacial [9]. Essas técnicas utilizam, dentre outras informações, o conhecimento da posição das fontes transmissoras em relação ao receptor. O desenvolvimento de técnicas para localização de fonte se insere nesse contexto. Aplicações comerciais utilizando múltiplas antenas de recepção, principalmente no *uplink* devem se tornar cada vez mais comuns com o aumento de banda utilizada e da taxa de transmissão. Alguns exemplos disso são os novos padrões para evolução da telefonia celular de terceira geração, conhecidos como LTE (*long-term evolution*). A utilização de múltiplas antenas nesses padrões está prevista para o *uplink* do LTE [10]. Técnicas para localização espacial de fontes também são muito importantes em aplicações que utilizam compartilhamento do acesso ao meio por SDMA (*space division multiple access*) [11]. O ajuste de filtros

espaciais para separar dados de usuários diferentes no SDMA é feito a partir das informações de localização desses usuários. Isso permite a todos os usuários transmitirem ao mesmo tempo usando a mesma banda, representando uma alternativa ao CDMA (*code division multiple access*) nessa situação. Os sistemas de localização de fonte, no entanto, devem ser adaptados ao contexto SDMA, devido à presença de múltiplos percursos. Para tal situação, são indicados algoritmos como o apresentado em [12].

Além da CCI, outro efeito negativo do aumento de taxas de transmissão é o aumento da ISI, que gera grande degradação na qualidade de serviço de um sistema de comunicação. Um dos métodos utilizados para o combate à ISI é o emprego de múltiplas portadoras de transmissão. O sistema conhecido como OFDM (*orthogonal frequency division multiplexing*) utiliza transformadas do tipo DFT (*discrete Fourier transform*) para gerar ortogonalidade entre portadoras na transmissão, possibilitando o uso de equalizadores com um único coeficiente, ou seja, de baixa complexidade computacional, para eliminar completamente a ISI, quando se utiliza um prefixo cíclico com dimensões apropriadas. Nos sistemas OFDM, a presença de alguns tipos de interferência, como o desvio na frequência central em portadoras (CFO, *carrier frequency offset*), causam perda dessa ortogonalidade. É então necessário que o efeito de CFO seja estimado e compensado no receptor. A utilização comercial do OFDM é encontrada tanto para TV digital, como no caso do padrão europeu DVB (*digital video broadcasting*), quanto para telefonia móvel, nos padrões LTE [13].

De certa forma, as estimações de DoA e CFO apresentam uma característica em comum, ambas representam a estimação de efeitos de modulação sobre a fase do sinal recebido, possibilitando que os mesmos métodos de estimação paramétricas possam ser adaptados a ambos cenários.

### 1.3 Objetivos

O objetivo principal deste trabalho é o desenvolvimento e a utilização de técnicas paramétricas de estimação que sejam alternativas de menor complexidade computacional em relação a algumas técnicas consagradas, em alguns cenários de sistemas

de comunicações.

Outros objetivos são a adaptação dessas técnicas para outros ambientes de sistemas de comunicações e o desenvolvimento de um *framework* unificado, que englobe todos os ambientes utilizados a fim de se gerar novos entendimentos sobre os algoritmos e favorecer a verificação de interrelações entre os algoritmos propostos.

A Figura 1.1 apresenta as relações entre os algoritmos desenvolvidos no doutorado. As setas indicam como se deu o fluxo de desenvolvimento dos algoritmos, com o objetivo de mostrar qual algoritmo contribuiu para o desenvolvimento do seguinte. Ao longo da tese são apresentados os seguintes algoritmos:

- O primeiro algoritmo desenvolvido ao longo do doutorado, CB-DoA (*covariance-based direction-of-arrival estimation*) se originou no ambiente de estimação de *direction-of-arrival* (DoA) em receptores com múltiplas antenas. O algoritmo CB-DoA está representado pelo bloco 1 na Figura 1.1;
- Em seguida, foi desenvolvido o algoritmo CB-CFO (*covariance-based CFO estimation*) utilizando técnicas semelhantes ao CB-DoA num cenário de estimação do *offset* das portadoras em sistemas OFDM. É mostrado no bloco 4 na Figura 1.1;
- Foram então desenvolvidos variantes do algoritmo CB-DoA que proporcionam menor complexidade computacional em relação tanto ao CB-DoA original quanto a suas contrapartidas do tipo ESPRIT (*estimation of parameters via rotational invariance techniques*) [14]. Tais algoritmos são representados nos blocos 2 e 3 na Figura 1.1;
- Foi desenvolvido um algoritmo para estimação de DoA que visa o processamento dos dados recebidos sem o uso de matrizes de covariância. Esse algoritmo, denominado DD-DoA, está representado no bloco 5;
- Por fim foi desenvolvida uma versão *Unitary* ESPRIT para estimação de CFO, representado pelo bloco 6.

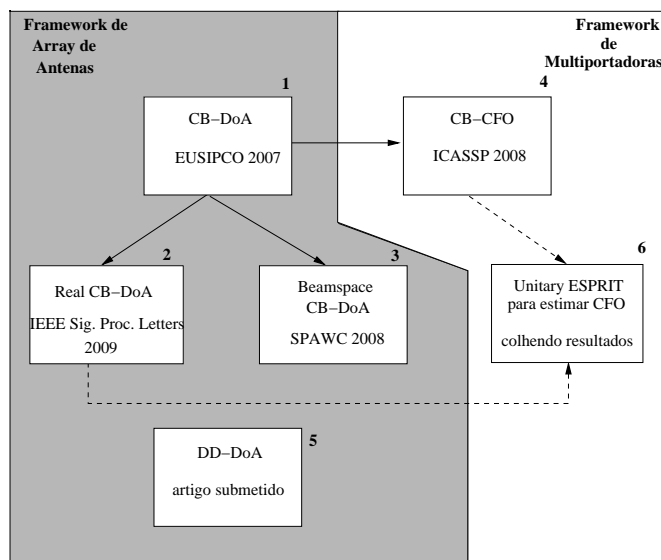


Figura 1.1: Esquema com os algoritmos apresentados na tese.

## 1.4 Sobre as Variáveis Utilizadas

Neste documento, procurou-se utilizar uma notação que fosse de uso comum, particularmente inspirada em [15]. Caracteres matemáticos sem negrito, por exemplo  $a$ , são usados para designar variáveis escalares. Vetores são representados através de caracteres minúsculos em negrito, por exemplo  $\mathbf{a}$ , enquanto matrizes são mostradas como caracteres maiúsculos em negrito, por exemplo  $\mathbf{A}$ . Não é feita uma distinção na notação entre variáveis determinísticas e aleatórias, deixando tal tarefa para explicações no corpo do texto. O operador  $\hat{\cdot}$  sobre uma variável indica que está sendo usada sua estimativa, por exemplo,  $\hat{a}$  indica a estimativa da variável  $a$ .

As variáveis aleatórias utilizadas são supostas como possuindo média zero, hipótese importante para o desenvolvimento dos algoritmos. A suposição de média zero pode ser retirada, porém são necessárias adaptações em alguns passos dos algoritmos desenvolvidos. Os processos aleatórios utilizados, além de também possuírem a suposição de média zero, também são supostos ergódicos e estacionários no sentido amplo. A única exceção à estacionaridade no sentido amplo é o TXK (Tong-Xu-Kailath) [16] em que várias variáveis são supostas como cicloestacionárias no sentido amplo, como será mostrado na Subseção 2.2.2.

Em algumas partes do texto, que envolvem descrição operacional dos algoritmos, foi escolhida uma notação de manipulação de matrizes inspirada na linguagem de *scripts* do Matlab.

Nas descrições dos algoritmos será utilizado um canal modelado como digital, da mesma maneira que a modelagem feita em [17]. Apenas na Subseção 2.3.2 é feita uma modelagem analógica do canal. A modelagem digital do canal está associada a uma taxa de amostragem que garante que seja obedecido o teorema de Nyquist, o que gera uma recepção sem *aliasing*.

## 1.5 Organização do Trabalho

No capítulo 2 são apresentados algoritmos clássicos para estimação paramétrica em sistemas de comunicações, em especial os algoritmos TXK e ESPRIT. As ideias contidas no TXK e no ESPRIT inspiraram o desenvolvimento dos algoritmos propostos. São também apresentadas no Capítulo 2 as principais famílias de algoritmos de estimação para um cenário com diversidade espacial. Nesse mesmo capítulo é abordado o método *Matrix Pencil* para estimação de parâmetros em somas de senóides [18].

No Capítulo 3 são apresentados novos algoritmos desenvolvidos para o cenário de estimação de DoA que são alternativas de baixa complexidade em comparação com o ESPRIT. Esses algoritmos estão representados na área cinza da Figura 1.1.

No Capítulo 4 são apresentados o problema de estimação de CFO para sistemas OFDM e o algoritmo ESPRIT [6] para estimação de CFO. Em tal contexto são propostos dois novos algoritmos com complexidade reduzida em relação ao ESPRIT. Tais algoritmos estão representados na área branca da Figura 1.1. No capítulo 4, também é construído um cenário unificado para alguns métodos propostos para estimação, como o CB-DoA e CB-CFO. Nesse mesmo cenário é desenvolvida uma versão generalizada do TXK englobando diferentes cenários de estimação.

No Capítulo 5, as principais conclusões da tese e as contribuições do trabalho são listadas. São também apresentadas propostas para desenvolvimentos de trabalhos futuros.

No Apêndice A é apresentada uma dedução matemática para expressões de MSE para dois dos algoritmos propostos.

# Capítulo 2

## Algoritmos Existentes para Equalização e Estimação de DoA

### 2.1 Introdução do Capítulo

Tradicionalmente, o combate à ISI pode ser feito com técnicas de processamento de sinais no receptor (equalização, ou precodificação como o método de Tomlinson-Harashima [17]), através da inserção de redundâncias de informação no transmissor (codificação de canal), ou de mudanças na banda do sinal transmitido através de técnicas de modulação (sistemas multiportadora), ou ainda por técnicas específicas de formatação de pulso [17].

Os primeiros sistemas de equalização utilizavam algum conhecimento da função de transferência do canal, seja através de um sistema externo para identificação do canal ou pelo envio de sequências de treinamento [17].

Em alguns cenários nos quais são inseridos sistemas de comunicação, tanto o uso de sistemas de identificação *a priori* do canal, como o envio de sequências de treinamento, não são ferramentas viáveis. Isso ocorre, por exemplo, em sistemas de espionagem ou no caso de testes não-invasivos. Em outras situações, apesar de ser possível o envio de sequências de treinamento, o uso de um período de treinamento gera um aumento na utilização de banda que pode ser considerado demasiado ou desnecessário. Esses casos levaram ao surgimento de algoritmos de equalização chamados cegos, os quais não necessitam de sequências de treinamento para o seu funcionamento [19].



Os primeiros algoritmos desenvolvidos com o conceito de equalização cega eram baseados em momentos e cumulantes de terceira e quarta ordens do sinal recebido. Isso gerava a presença incômoda de mínimos locais e, na maioria dos casos, uma convergência lenta na equalização [19] [20]. Por exemplo, o algoritmo CMA (*constant-modulus algorithm*) se aproveita da presença de símbolos com módulo constante na constelação utilizada.

Posteriormente, o desenvolvimento de técnicas que utilizam propriedades ligadas à cicloestacionaridade do sinal recebido [21] levou ao surgimento de equalizadores cegos com base apenas em estatísticas de segunda ordem do sinal recebido. O primeiro método de equalização cega que utiliza apenas estatísticas de segunda ordem é denominado TXK (Tong, Xu, Kailath) [16], e baseia-se em manipulações de matrizes de covariância do sinal recebido. Essa técnica será descrita na Subseção 2.2.2 e serviu de inspiração para o desenvolvimento dos algoritmos propostos, descritos nos Capítulos 3 e 4.

O desenvolvimento de sistemas de transmissão sem fio com múltiplas antenas de transmissão e recepção contribuiu para o uso mais eficiente do espectro disponível e para o combate mais eficiente das interferências inerentes aos sistemas de transmissão [9] [22]. O uso de sistemas com múltiplas antenas levou ao surgimento de técnicas de localização espacial de fontes, como visto no Capítulo 1. Técnicas de localização de fonte inicialmente se basearam em métodos não-paramétricos de estimação, os quais, por sua própria natureza, caracterizam-se em geral por apresentar alta complexidade computacional.

Técnicas de estimação paramétricas, derivadas a partir do estimador ML, foram inicialmente desenvolvidas aproveitando-se características conhecidas do espectro a ser estimado, a fim de diminuir a complexidade em relação aos métodos não-paramétricos.

O desenvolvimento do método MUSIC (*multiple signal classification*) [23] representa uma alternativa a métodos baseados em ML, com menor complexidade computacional. O uso do MUSIC, em sua versão original *spectral* MUSIC descrita em [23], não requer nenhuma suposição sobre a geometria das antenas de recepção, garantindo uma certa liberdade no projeto do receptor.

O método ESPRIT (*estimation of parameters via rotational invariance techni-*

ques) [14] [24] levou a uma redução ainda mais significativa na complexidade computacional ao impor algumas restrições sobre a geometria do *array* de antenas de recepção.

Neste capítulo é inicialmente apresentado o algoritmo TXK. A seguir é apresentado o ambiente de estimação de DoA, além do método clássico ESPRIT. Então é apresentada a técnica *beamspace*, que permite uma diminuição ainda mais acentuada na complexidade computacional dos algoritmos de estimação paramétrica [25]. São discutidos alguns tópicos ligados ao uso do *beamspace*, em comparação com os métodos tradicionais, aplicados no chamado *elementspace*. É então mostrado o algoritmo *Unitary ESPRIT*, que possibilita a utilização apenas de operações reais. Por fim, é mostrado o algoritmo *Matrix Pencil* (MP), que tem uma formulação parecida com os algoritmos propostos.

Os algoritmos clássicos apresentados neste capítulo serviram de inspiração para os algoritmos propostos ao longo desta tese.

## 2.2 Equalização Cega

Uma das técnicas cujas ideias serviram de inspiração para o desenvolvimento dos algoritmos propostos nesta tese é um algoritmo originário do cenário de equalização cega. Tal algoritmo foi denominado TXK [16], um acrônimo envolvendo as iniciais de seus autores. Para facilitar o entendimento do desenvolvimento dos algoritmos propostos nos Capítulos 3 e 4, o ambiente de equalização cega assim como o algoritmo TXK serão mostrados nesta seção.

### 2.2.1 Descrição do Ambiente

Considere o ambiente de comunicação digital associado ao processo de equalização cega representado pela Figura 2.1, onde  $s(k)$  representa o sinal transmitido,  $n(k)$  é o ruído branco, aditivo, com distribuição gaussiana (AWGN, *additive white Gaussian noise*),  $x(k)$  refere-se ao sinal recebido e  $\hat{s}(k)$  representa uma estimação da sequência  $s(k)$ .

Representado-se o canal invariante no tempo por sua resposta ao impulso  $h(k)$ ,

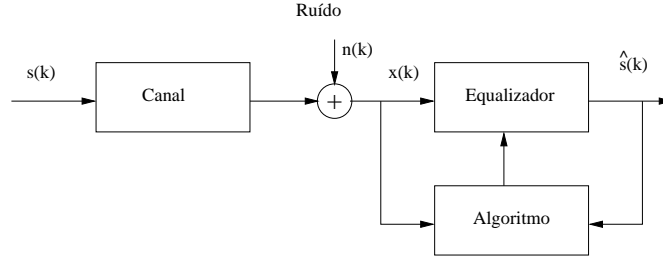


Figura 2.1: Ambiente de comunicações associado ao processo de equalização cega.

de comprimento  $L$ , observa-se que

$$x(k) = \sum_{i=0}^{L-1} s(i)h(k-i) + n(k). \quad (2.1)$$

No problema de equalização cega, deseja-se que a estimativa  $\hat{s}(k)$  esteja o mais próximo possível de  $s(k)$ , sem conhecimento prévio a respeito tanto de  $s(k)$  quanto de  $h(k)$ . Há apenas conhecimentos estatísticos a respeito de  $s(k)$ . Considera-se que tanto  $s(k)$  quanto  $n(k)$  sejam gerados por processos estocásticos de média zero com amostras decorrelacionadas no tempo. Não há também conhecimento sobre o ruído, apenas que suas amostras são decorrelacionadas.

Utilizando-se propriedades ligadas à cicloestacionaridade, o sinal de entrada é sobreamostrado a uma taxa constante no TXK, considerada aqui  $P = 1/\alpha$ , com  $\alpha < 1$ , vezes a taxa original, sendo  $\alpha$  inteiro. Considerando o sinal originalmente estacionário, quando medido na taxa de símbolos. O sinal sobreamostrado pode ser considerado um sinal cicloestacionário [21], com período  $\alpha$  vezes o período original. Usando a teoria de representação em séries de translação (TSR, *translational series representation*) [21], formulada por Gardner em [21] para sinais cicloestacionários, pode-se modelar alternativamente como se cada conjunto de subamostras com período  $\alpha$  passasse por um subcanal diferente. Esses subcanais são modelados como sendo independentes entre si, cada um com uma resposta ao impulso que não está relacionada à dos demais. Considerando-se que cada uma das  $M$  TSRs seja modelada como tendo comprimento  $Q$  cujo valor engloba o comprimento do canal utilizado, a fim de se facilitar uma representação vetorial. Com isso, para a representação do sinal recebido  $x(k)$ , a equação (2.1) para um sistema sobreamostrado

se torna

$$x(k\alpha + m) = \sum_{i=0}^{Q-1} s(i + m)h(k\alpha - i - m) + n(k\alpha + m), \quad 0 \leq m < M, \quad (2.2)$$

onde  $m$  representa o índice da TSR em questão. Definem-se então os seguintes vetores,

$$\mathbf{s}(k) = \begin{bmatrix} s(kQ) \\ s(kQ + 1) \\ \vdots \\ s((k + 1)Q - 1) \end{bmatrix}, \quad \mathbf{x}(k) = \begin{bmatrix} x(kP) \\ x(kP + 1) \\ \vdots \\ x((k + 1)P - 1) \end{bmatrix}. \quad (2.3)$$

Para o funcionamento do algoritmo,  $P$  deve obedecer  $P > Q$ . Define-se também uma matriz auxiliar  $\mathbf{H}$ , contendo a resposta ao impulso dos múltiplos canais gerados como consequência da representação em TSRs,

$$\mathbf{H} = \begin{bmatrix} h(k - L) & h(k - L - 1) & \dots & h(k - L - Q + 1) \\ h(2k - L) & h(2k - L - 1) & \dots & h(2k - L - Q + 1) \\ \vdots & \vdots & \ddots & \vdots \\ h(Pk - L) & h(Pk - L - 1) & \dots & h(Pk - L - Q + 1) \end{bmatrix}. \quad (2.4)$$

Modela-se ainda o ruído como um conjunto de TSRs,

$$\mathbf{n}(k) = \begin{bmatrix} n(kP) \\ n(kP + 1) \\ \vdots \\ n((k + 1)P - 1) \end{bmatrix}, \quad (2.5)$$

chegando-se então a uma modelagem vetorial para o sistema,

$$\mathbf{x}(k) = \mathbf{H}\mathbf{s}(k) + \mathbf{n}(k), \quad (2.6)$$

onde o ruído vetorial continua sendo considerado branco. É mostrado em [21] que, caso  $s(k)$  e  $n(k)$  sejam originalmente cicloestacionários quando sobreamostrados,  $\mathbf{s}(k)$ ,  $\mathbf{n}(k)$  e  $\mathbf{x}(k)$  serão vetores estacionários nas condições apresentadas nesta seção. Com isso, um sistema com sinais escalares cicloestacionários é transformado num sistema com sinais vetoriais estacionários. Isso permite a utilização, no sistema modelado na equação (2.6), de ferramentas matemáticas tradicionalmente usadas para sinais estacionários. Em comparação com o sistema descrito pela equação (2.1),

o sistema representado pela equação (2.6) possui mais redundâncias que o sistema original, como veremos a seguir. Essa redundância permite o uso de algoritmos de equalização cega com base em momentos de 2ª ordem do sinal recebido.

## 2.2.2 Algoritmo TXK para Equalização Cega

**Apresentação das Estruturas:** Considere as seguintes matrizes de covariância

$$\mathbf{R}_s(\tau) = E[\mathbf{s}(k)\mathbf{s}^H(k - \tau)], \quad (2.7)$$

$$\mathbf{R}_x(\tau) = E[\mathbf{x}(k)\mathbf{x}^H(k - \tau)]. \quad (2.8)$$

Considerando-se a entrada branca, tem-se

$$\mathbf{R}_s(\tau) = \begin{cases} \mathbf{L}^\tau, & \tau > 0, \\ \mathbf{I}, & \tau = 0, \\ (\mathbf{L}^H)^{|\tau|}, & \tau < 0, \end{cases} \quad (2.9)$$

onde  $\mathbf{I}$  corresponde a uma matriz identidade de dimensões adequadas, e

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}. \quad (2.10)$$

A partir das equações (2.6) e (2.9), tem-se que

$$\mathbf{R}_x(\tau) = \mathbf{H}\mathbf{L}^\tau\mathbf{H}^H + \sigma_N^2\mathbf{L}_{\text{TXK}}, \quad \tau \geq 0, \quad (2.11)$$

onde  $\sigma_N^2$  representa a potência do ruído e  $\mathbf{L}_{\text{TXK}}$  é definido como

$$\mathbf{L}_{\text{TXK}} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}. \quad (2.12)$$

**Teorema de Identificação:** No artigo [16], no qual o algoritmo TXK é proposto, é apresentado um teorema importante sobre identificação de canal. Sejam uma matriz de canal  $\mathbf{H}$ , com posto completo por colunas com valor  $Q$ , e o vetor  $\mathbf{s}$  de dados gerado por um processo aleatório branco, de média zero. O vetor  $\mathbf{s}$  é transmitido através do canal  $\mathbf{H}$  e corrompido por  $\mathbf{n}$ , gerando-se o sinal recebido  $\mathbf{x}$ , como mostrado na equação (2.6). Então o canal representado por  $\mathbf{H}$  é unicamente identificado pela autocovariância  $\mathbf{R}_x(0)$  de  $\mathbf{x}$  com *lag*  $\tau = 0$  e a autocovariância  $\mathbf{R}_x(1)$  de  $\mathbf{x}$  com *lag*  $\tau = 1$ . A prova do teorema encontra-se no artigo [16].

Uma consequência deste teorema é que a estimação do canal gerada a partir das propriedades de cicloestacionaridade é realizada mesmo quando se apresenta uma ambiguidade de fase, ou seja, considerando-se  $\hat{\mathbf{H}}$  como a estimativa do canal, então

$$\hat{\mathbf{H}} = \mathbf{H}\mathbf{Q}_1, \quad (2.13)$$

onde  $\mathbf{Q}_1$  é uma matriz unitária que obedece  $\mathbf{Q}_1 = e^{j\phi}\mathbf{I}$ . Para corrigir tal distorção de fase, deve-se estimar  $\phi$  e em seguida efetuar a operação

$$\mathbf{H} = e^{-j\phi}\hat{\mathbf{H}}. \quad (2.14)$$

Como o canal deve ser unicamente identificado a partir de  $\mathbf{R}_x(1)$ , então mesmo que haja uma ambiguidade de identificação deve ser respeitada a seguinte equação

$$\mathbf{R}_x(1) = \mathbf{H}\mathbf{L}\mathbf{H}^H = \mathbf{H}\mathbf{Q}_1\mathbf{L}\mathbf{Q}_1^H\mathbf{H}^H. \quad (2.15)$$

De maneira equivalente, a condição de identificação dada por (2.15) é equivalente a

$$\mathbf{L} = \mathbf{Q}_1\mathbf{L}\mathbf{Q}_1^H. \quad (2.16)$$

Considerando  $\mathbf{q}_{1,i}$  como a  $i$ -ésima coluna de  $\mathbf{Q}_1$ , então

$$\mathbf{L}\mathbf{q}_{1,i} = \mathbf{q}_{1,i+1}, \quad (2.17)$$

que é uma equação característica de uma cadeia de Jordan [16].

**Processamento sobre os Dados Recebidos:** Faz-se uma decomposição de  $\mathbf{R}_x(0)$  em autovalores e autovetores, (EVD, *eigenvalue/eigenvector decomposition*) ou seja,

$$\mathbf{R}_x(0) = \mathbf{U}\mathbf{\Lambda}^2\mathbf{U}^H, \quad (2.18)$$

já que  $\mathbf{R}_x(0)$  é uma matriz de covariância, sendo, portanto, hermitiana e positiva semidefinida [16]. Ordenando-se os autovalores  $\lambda_i$ , com  $0 \leq i \leq (P - 1)$  de  $\mathbf{R}_x(0)$  em ordem decrescente, pode-se escrever que

$$\lambda_i = \begin{cases} \sigma_N^2 + \delta_i, & 0 \leq i \leq (Q - 1), \\ \sigma_N^2, & Q \leq i \leq (P - 1). \end{cases} \quad (2.19)$$

Os autovetores correspondentes aos  $Q$  maiores autovalores geram um subespaço conhecido como subespaço do sinal (mas que contém sinal e ruído apesar da nomenclatura), enquanto os demais autovetores geram o chamado subespaço do ruído. Então, formam-se a matriz diagonal  $\mathbf{\Lambda}_s^2$ , contendo os autovalores do subespaço do sinal de  $\mathbf{R}_x(0)$  em sua diagonal, e  $\mathbf{U}_s$ , contendo os correspondentes autovetores em suas colunas. Considere também a matriz auxiliar  $\mathbf{F}$ , de dimensões  $Q \times P$ , definida por

$$\mathbf{F} = \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H, \quad (2.20)$$

o que de certa forma pode ser visto como um processo de branqueamento.

A partir da equação (2.11), percebe-se que

$$\mathbf{R}_x(1) = \mathbf{H} \mathbf{L} \mathbf{H}^H + \sigma_N^2 \mathbf{L}_{\text{TXK}}. \quad (2.21)$$

A estrutura

$$\mathbf{R}_a = \mathbf{R}_x(1) - \sigma_N^2 \mathbf{L}_{\text{TXK}} = \mathbf{H} \mathbf{L} \mathbf{H}^H \quad (2.22)$$

é denominada *pencil* matricial (MP, *matrix pencil*).

As operações a seguir têm como característica a projeção de  $\mathbf{R}_a$  no subespaço dos sinais, obtido a partir da equação (2.21). Isso gera uma matriz que possui uma estrutura conhecida, a partir da qual são obtidas estimações  $\hat{\mathbf{s}}(k)$ . Considere agora a matriz auxiliar

$$\begin{aligned} \mathbf{\Psi}_{\text{TXK}} &= \mathbf{F} \mathbf{R}_a \mathbf{F}^\dagger \\ &= (\mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H) \mathbf{R}_a (\mathbf{U}_s \mathbf{\Lambda}_s) \\ &= (\mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H) \mathbf{H} \mathbf{L} \mathbf{H}^H (\mathbf{U}_s \mathbf{\Lambda}_s). \end{aligned} \quad (2.23)$$

Aplica-se uma decomposição em valores singulares (SVD, *singular value decomposition*) em  $\mathbf{\Psi}_{\text{TXK}}$ , de forma que

$$\mathbf{\Psi}_{\text{TXK}} = \mathbf{Y} \mathbf{\Sigma}_s \mathbf{Z}^H. \quad (2.24)$$

Até o momento foi feita uma estimação dos parâmetros  $\Psi$ . A seguir, esses parâmetros serão mapeados nas variáveis de interesse, ou seja, o conjunto de dados transmitidos e uma estimação para o canal.

**Utilizando o Teorema:** Neste ponto, torna-se importante utilizar o teorema de identificação mostrado em [16], apresentado no início desta seção. Usando a última coluna  $\mathbf{y}$  da matriz de vetores singulares à esquerda  $\mathbf{Y}$ , gera-se uma nova matriz a partir da equação (2.17), isto é, a partir do conceito de cadeias de Jordan [16],

$$\mathbf{Q}_2 = \begin{bmatrix} \mathbf{y} & \Psi_{\text{TXK}\mathbf{Y}} & \dots & \Psi_{\text{TXK}\mathbf{Y}}^{P-1} \end{bmatrix}. \quad (2.25)$$

Uma estimativa  $\hat{\mathbf{H}}$  da matriz de canal pode ser obtida com uma incerteza de fase  $\phi$ ,

$$\hat{\mathbf{H}} = \mathbf{U}_s \Lambda_s \mathbf{Q}_2 e^{j\phi}. \quad (2.26)$$

No entanto, a partir do que foi estabelecido no teorema de identificação, a equalização *zero forcing* (ZF) [17] é obtida sem incerteza de fase,

$$\hat{\mathbf{s}}(k) = \mathbf{Q}_2^H \mathbf{F}\mathbf{x}(k). \quad (2.27)$$

Na Tabela 2.1, o algoritmo TXK encontra-se descrito de maneira resumida, supondo-se já estimadas as estruturas de covariância  $\mathbf{R}_x(0)$  e  $\mathbf{R}_x(1)$ . Foi utilizada uma notação com base na linguagem de *scripts* do Matlab, em que  $\mathbf{Y}(:, \text{end})$  representa a última coluna da matriz  $\mathbf{Y}$ .

**Transformação Bilateral:** A partir do que foi visto na equação (2.23) do algoritmo TXK, é definida aqui a operação de transformação bilateral (TB) de maneira que

$$[\Psi] = \text{TB}(\mathbf{U}_s, \Lambda_s, \mathbf{R}_a) \quad (2.28)$$

Nessa operação, a matriz  $\mathbf{R}_a$  é transformada em ambos os lados utilizando a matriz definida por  $\Lambda_s^{-1} \mathbf{U}_s^H$ ,

$$\Psi_{\text{TXK}} = \Lambda_s^{-1} \mathbf{U}_s^H \mathbf{R}_a (\Lambda_s^{-1} \mathbf{U}_s^H)^\dagger = \Lambda_s^{-1} \mathbf{U}_s^H \mathbf{R}_a \mathbf{U}_s^H \Lambda_s. \quad (2.29)$$

Na Figura 2.2 está representado o processo de estimação do algoritmo TXK.



Tabela 2.1: Resumo das operações do algoritmo TXK.

<b>TXK</b>
$[\mathbf{U}_s, \Lambda_s^2] = \text{EVD}(\hat{\mathbf{R}}_x(0) - \sigma^2 \mathbf{I})$
$\mathbf{F} = \Lambda_s^{-1} \mathbf{U}_s^H$
$\mathbf{R}_a = \hat{\mathbf{R}}_x(1) - \hat{\sigma}^2 \mathbf{L}$
$\Psi = \mathbf{F} \mathbf{R}_a \mathbf{F}^\dagger$
$[\mathbf{Y}] = \text{SVD}(\Psi)$
$\mathbf{y} = \mathbf{Y}(:, \text{end})$
$\mathbf{Q}_2 = [\mathbf{y} \quad \Psi \mathbf{y} \quad \dots \quad \Psi^{P-1} \mathbf{y}]$
$\hat{\mathbf{s}}(k) = \mathbf{Q}_2^H \mathbf{F} \mathbf{x}(k)$

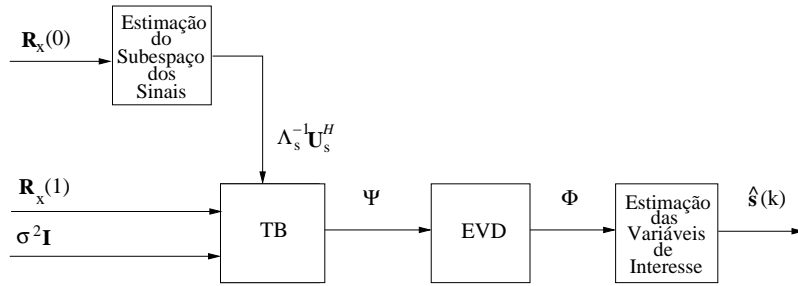


Figura 2.2: Diagrama de blocos representando o processo de estimação do algoritmo TXK.

## 2.3 Estimação de DoA

Os algoritmos desenvolvidos nesta tese foram inicialmente aplicados para estimação de DoA através de um *array* de antenas de recepção. Portanto, nesta seção será caracterizado um sistema de comunicações que utiliza propriedades ligadas à diversidade espacial para separação de usuários.

### 2.3.1 Métodos Paramétricos

Nesta subseção serão apresentadas as principais famílias de métodos de estimação de parâmetros para um ambiente de estimação de DoA. Utilizando-se uma classificação simplificada, a estimação de sinais pode ser realizada: ou através da estimação da forma de onda completa do sinal original, o que caracteriza um *método não-paramétrico*; ou através da estimação de parâmetros, quando se tem algum conhecimento sobre a forma de onda original, o que caracteriza uma *estimação paramétrica*. Por sua vez, os métodos paramétricos inicialmente se basearam no critério ML. No entanto, com o intuito de diminuir sua complexidade computacional, foram desenvolvidos métodos subótimos para o critério ML. Esses métodos podem ser classificados da seguinte maneira [26]:

- **Métodos do tipo *subspace fitting***: Têm como objetivo a minimização de funções-custo do tipo  $\|\mathbf{y} - \mathbf{A}\Phi\mathbf{s}\|_2^2$ , onde  $\mathbf{y}$  é a variável observável na recepção,  $\mathbf{s}$  é o sinal transmitido,  $\mathbf{A}$  representa uma matriz de mistura,  $\Phi$  é uma redundância na estrutura da matriz de mistura e  $\|\cdot\|_F$  representa a norma de Frobenius de uma matriz [27]. Exemplos destes métodos incluem os algoritmos MUSIC [28] e *Matrix Pencil* (MP)[18];
- **Métodos do tipo *single shift-invariance* (SSI)**: Utilizam manipulações da solução da equação de invariância,  $\mathbf{E}_x = \mathbf{E}_y\mathbf{\Psi}_{\text{SSI}}$ , onde  $\mathbf{E}_x$  e  $\mathbf{E}_y$  são matrizes ligadas ao subespaço dos sinais e  $\mathbf{\Psi}_{\text{SSI}}$  é a chamada matriz de invariância que representa uma redundância na representação dos sistemas. Exemplo deste tipo de método é o algoritmo ESPRIT [14];
- **Métodos do tipo *orthogonal vector***: Baseiam-se na procura de vetores ortogonais ao subespaço do ruído. Como exemplos, têm-se os algoritmos de Kumaresan-Tufts [29] e *Root-MUSIC* [30].

Os métodos propostos, a serem apresentados no Capítulo 3, pertencem à classe de métodos do tipo SSI, apresentando uma solução alternativa para a resolução da equação de invariância.

Uma outra classificação é mostrada em [31], onde os métodos de estimação são divididos em ML, predição linear e baseados em subespaços. Os métodos basea-

dos em subespaço são subdivididos em *orthogonal vector* e *subspace fitting*, com os métodos SSI dentro do tipo *subspace fitting*.

Estendendo-se a classificação proposta em [32], os métodos do tipo SSI apresentam 4 etapas de funcionamento, conforme ilustrado na Figura 2.3,

**SSI1 Identificação das dimensões do subespaço dos sinais:** Como o número de fontes de transmissão pode ser desconhecido ou variar ao longo do tempo, pode ser necessária a identificação do número de fontes utilizadas, através de métodos como os *Akaike's information-theoretic criteria* (AIC) [33].

**SSI2 Estimação do subespaço dos sinais:** Esta etapa é tradicionalmente feita através das operações EVD ou SVD. Essas operações são custosas computacionalmente. Com isso, alguns métodos de *subspace tracking*, como o PASTd (*projection approximation subspace tracking*) [34], podem ser utilizados como aproximação neste processo de estimação de subespaço.

**SSI3 Solução da equação de invariância:** A propriedade de invariância ao deslocamento é crucial para a realização da estimação, pois fornece a redundância de informação necessária para sua solução. A solução da chamada equação de invariância  $\mathbf{E}_x = \mathbf{E}_y \Psi_{\text{SSI}}$  pode ser realizada através de técnicas como LS (*least-squares*) ou TLS (*total least-squares*) [27].

**SSI4 Identificação de parâmetros do espectro espacial:** Esta é a etapa final dos métodos utilizando em SSI, que consiste numa manipulação matemática da solução obtida para a equação de invariância para se obter a estimativa desejada. A partir do parâmetro estimado, obtém-se também uma estimação da variável de ambiente desejada, por exemplo, a estimação de DoA.

Todos os algoritmos a serem apresentados nos Capítulos 3 e 4 consistem numa alternativa à etapa 3, mais especificamente são métodos alternativos ao ESPRIT e suas variantes. Por causa dessas alterações na realização da etapa SSI3, operações nas etapas anteriores requerem estruturas de dimensões menores que as de algoritmos tradicionais, como o ESPRIT, caracterizando uma menor complexidade computacional também na etapa SSI2. Há uma relação próxima entre os algoritmos desenvolvidos e os métodos do tipo MP para estimação de senóides em ruído branco

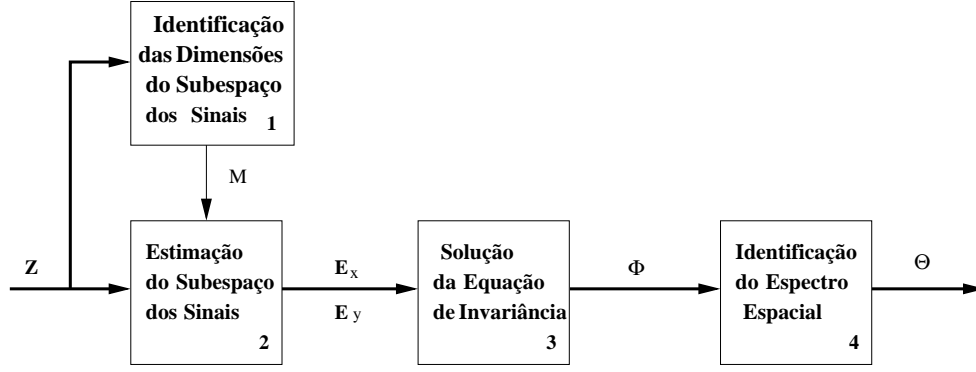


Figura 2.3: Etapas dos métodos SSI.

(*harmonic retrieval*). As relações com algoritmos MP serão mais explicitadas na Subseção 2.4 e Seção 3.7.

Apesar de se apresentar nesta seção um *framework* envolvendo métodos de estimação apenas para sistemas com diversidade espacial, existem variações voltadas para a estimação de desvio da frequência da portadora em OFDM, como será mostrado na Seção 4.6.

Define-se uma função genérica SSI1 para a realização da primeira etapa de estimação, ou seja a identificação do número de fontes utilizadas, ou seja,

$$[\hat{M}] = \text{SSI1}(\mathbf{Z}), \quad (2.30)$$

onde  $\mathbf{Z}$  representa a matriz contendo em suas colunas todos os dados recebidos e  $M$  representa o número de fontes utilizado. A notação utilizada tem por base *scripts* Matlab, com o uso de colchetes para denotar a saída.

Além disso, define-se uma função SSI2 relacionada à segunda etapa dos métodos SSI,

$$[\mathbf{E}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_z, \mathbf{R}_n, \hat{M}), \quad (2.31)$$

onde  $\mathbf{R}_z$  representa uma estimativa  $\mathbf{Z}\mathbf{Z}^H$  para a covariância  $E[\mathbf{z}(k)\mathbf{z}^H(k)]$  dos dados recebidos e  $\mathbf{R}_n$  uma estimativa da matriz normalizada (sem potência) para covariância do ruído. Em SSI2, é feita uma decomposição do par  $\{\mathbf{R}_z, \mathbf{R}_n\}$  em autovalores e autovetores generalizados (GEVD - *generalized eigenvalue/eigenvector decomposition*),

$$\text{GEVD}(\mathbf{R}_z, \mathbf{R}_n) = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H. \quad (2.32)$$

O uso de uma GEVD é necessária quando o ruído não é branco, ou seja, sua autocovariância não é uma matriz identidade. Ordenando-se de maneira decrescente os

autovalores generalizados  $\lambda_i$ , percebe-se a seguinte relação

$$\lambda_i = \begin{cases} \sigma_N^2 + \delta_i, & 0 \leq i < M, \\ \sigma_N^2, & M \leq i < 2N. \end{cases}, \quad (2.33)$$

onde  $\sigma_N^2$  representa a potência do ruído. Os  $M$  autovetores generalizados correspondentes aos maiores autovalores generalizados formam o chamado subespaço do sinal, enquanto os restantes geram o subespaço do ruído. Considera-se  $\mathbf{U}_s$  como a matriz que contém em suas colunas os autovetores generalizados correspondentes ao subespaço dos sinais. É definida então a matriz  $\mathbf{E}_s$  tal que

$$\mathbf{E}_s = \mathbf{\Sigma}_n \mathbf{U}_s. \quad (2.34)$$

É definida uma variação SSI2b para a função SSI2, válida quando o ruído é branco,

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_z, M). \quad (2.35)$$

Seu funcionamento é idêntico ao de SSI2, com o uso de uma identidade  $\mathbf{I}$  no lugar de  $\mathbf{R}_n$ . Sua definição é importante pois a operação GEVD se transforma em EVD, muito menos complexa computacionalmente.

A função SSI3 corresponde a um dos algoritmos propostos ou aos algoritmos clássicos (como o ESPRIT). No caso do ESPRIT, a resolução da equação de invariância é feita através de métodos LS ou TLS, a partir de variáveis resultantes da SSI2. No caso dos algoritmos propostos, será usada a função TB, que aparece no contexto do TXK, na Subsecção 2.2.2, e, em seguida, uma EVD.

É definida uma função SSI4 que depende do ambiente de estimação utilizado. Sua expressão genérica é

$$[\boldsymbol{\theta}] = \text{SSI4}(\boldsymbol{\Phi}), \quad (2.36)$$

onde  $\boldsymbol{\theta}$  é um vetor contendo as estimações das variáveis desejadas. A seguir essa expressão será definida de acordo com o ambiente utilizado.

As funções genéricas definidas aqui serão utilizadas posteriormente na descrição dos algoritmos.

**Estimação TLS:** Em alguns dos algoritmos a serem apresentados, será utilizada uma estimação TLS. A estimação TLS apresentada aqui é mostrada no artigo [35].

Considere o sistema linear representado por

$$\mathbf{A}\mathbf{X} = \mathbf{B} + \mathbf{V}, \quad (2.37)$$

onde  $\mathbf{A}$  é uma matriz conhecida de dimensões  $m \times n$  e  $\mathbf{B}$  é uma matriz conhecida de dimensões  $m \times p$  e a matriz  $\mathbf{V}$  representa um ruído. Deseja-se estimar  $\mathbf{X}$  a partir de  $\mathbf{A}$  e  $\mathbf{B}$ . Considere a função

$$[\hat{\mathbf{X}}_{\text{TLS}}] = \text{TLS}(\mathbf{A}, \mathbf{B}). \quad (2.38)$$

Considere a matriz auxiliar

$$\mathbf{C} = \begin{bmatrix} \mathbf{A}^H \\ \mathbf{B}^H \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix}, \quad (2.39)$$

onde  $\mathbf{C}$  tem dimensões  $(n+p) \times (n+p)$ . Faz-se uma EVD sobre  $\mathbf{C}$ ,

$$\text{EVD}(\mathbf{C}) = \mathbf{E}\mathbf{D}\mathbf{E}^{-1}. \quad (2.40)$$

A matriz  $\mathbf{E}$  também equivale à matriz de vetores singulares à direita de  $\begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix}$ . Particiona-se  $\mathbf{E}$ , que possui dimensões  $(n+p) \times (n+p)$  em 4 submatrizes, da seguinte maneira

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}, \quad (2.41)$$

onde  $\mathbf{E}_{11}$  tem dimensões  $n \times n$ ,  $\mathbf{E}_{12}$  tem dimensões  $n \times p$ ,  $\mathbf{E}_{21}$  possui dimensões e  $\mathbf{E}_{22}$  tem comprimento  $p \times p$ . A estimação  $\hat{\mathbf{X}}_{\text{TLS}}$  é encontrada através de

$$\hat{\mathbf{X}}_{\text{TLS}} = -\mathbf{E}_{12}\mathbf{E}_{22}^{-1}, \quad (2.42)$$

desde que  $\mathbf{E}_{22}$  seja inversível.

### 2.3.2 Descrição do Ambiente

Num sistema com múltiplas fontes de transmissão e múltiplas antenas de recepção, considere as fontes posicionadas muito distantes dos receptores, ou seja, num *far-field* em relação ao *array* de recepção. Considera-se ainda que o meio de propagação seja isotrópico. Com isso, as ondas eletromagnéticas incidentes no *array* de recepção formam uma frente de onda aproximadamente plana [36].

Dado que as fontes de transmissão estão localizadas num mesmo plano, o problema de localização de fonte transforma-se num problema de se determinar o ângulo de recepção, ou seja, o DoA da frente de onda no *array* receptor.

Considere um sistema de comunicações com  $M$  antenas de transmissão e  $2N$  antenas de recepção, como representado na Figura 2.4, onde  $M < N$ . Considere também que seja realizada transmissão em banda estreita, e que o meio está imerso em AWGN. O conjunto receptor é restrito de tal forma que as antenas de recepção possam ser agrupadas em  $N$  grupos de 2 antenas, chamados de *doublets*, tal que a distância das antenas pertencentes a um dado *doublet* seja representada por um mesmo vetor *displacement*. *Displacement* é um vetor que tem como origem uma das antenas do *doublet*, e seu destino é a outra antena [14]. Pode-se generalizar a configuração do *array* de modo que uma mesma antena pertença a dois *doublets* distintos, desde que seja a origem do vetor *displacement* em um deles e o destino do vetor *displacement* em outro *doublet*. Nesse caso, o número total de antenas estaria entre  $N$  e  $2N$ .

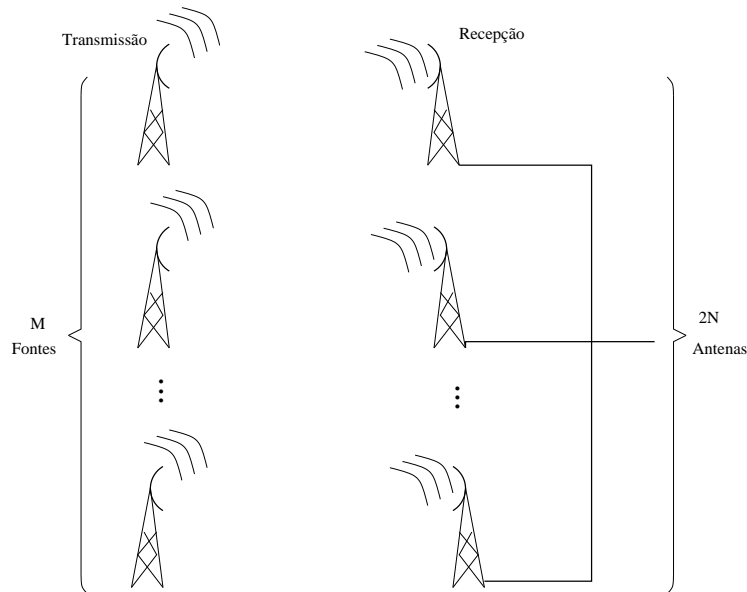


Figura 2.4: Ambiente de múltiplas fontes de transmissão e múltiplas antenas de recepção.

Seja  $s_m(t)$  o sinal transmitido pela  $m$ -ésima antena no tempo contínuo  $t$ , com  $0 \leq m \leq (M - 1)$  e sejam  $x_i(t)$  e  $y_i(t)$  os sinais recebidos no  $i$ -ésimo *doublet* no tempo  $t$ , com  $0 \leq i \leq (N - 1)$ . Além disso, o ângulo de chegada referente à  $m$ -ésima

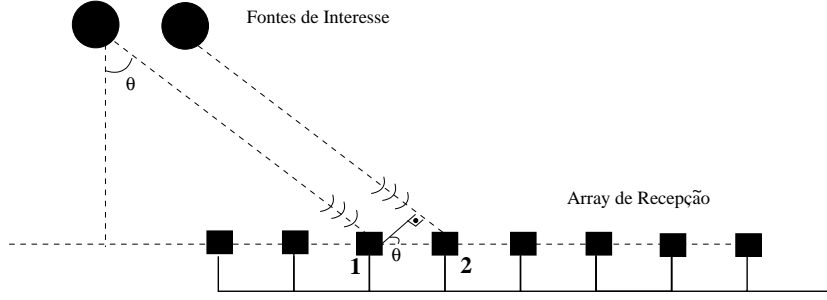


Figura 2.5: Ângulo  $\hat{\theta}$  medido por algoritmos para estimação de DoA.

fonte é referido como  $\theta_m$ . Já o ganho da  $i$ -ésima antena de recepção para o ângulo  $\theta_m$  é representado por  $a_i(\theta_m)$ . A medição do DoA  $\theta$  é feita como representado na Figura 2.5, onde os círculos representam fontes transmissoras enquanto os quadrados representam antenas receptoras e o ângulo é medido de acordo com a linha entre as antenas 1 e 2 pertencentes ao mesmo *doublet*. Por geometria, pode-se verificar que o DoA  $\theta$  corresponde ao ângulo transmitido em relação a uma linha perpendicular entre a fonte e o *array* de recepção, como também mostrado na Figura 2.5.

Os sinais  $x_i(t)$  e  $y_i(t)$  obedecem as seguintes relações

$$x_i(t) = \sum_{m=0}^{M-1} s_m(t) a_i(\theta_m) + n_{x,i}(t), \quad (2.43)$$

$$y_i(t) = \sum_{m=0}^{M-1} s_m(t) e^{\frac{j\omega\delta}{c} \sin(\theta_m)} a_i(\theta_m) + n_{y,i}(t), \quad (2.44)$$

onde  $c$  representa a velocidade da luz e  $n_{x,i}(t)$  e  $n_{y,i}(t)$  representam o ruído que incide nas duas antenas do  $i$ -ésimo *doublet*. Além disso, supôs-se que  $s_m(t)$  seja um sinal em banda estreita. A variável  $\omega$  representa a velocidade angular de propagação da onda transmitida, enquanto  $\delta$  é o módulo do vetor *displacement*.

Definem-se então as variáveis equivalentes no domínio do tempo discreto  $k$ ,

$$\mathbf{x}(k) = [x_0(k) \ x_1(k) \ \dots \ x_{N-1}(k)]^T, \quad (2.45)$$

$$\mathbf{y}(k) = [y_0(k) \ y_1(k) \ \dots \ y_{N-1}(k)]^T, \quad (2.46)$$

$$\mathbf{n}_x(k) = [n_{x,0}(k) \ n_{x,1}(k) \ \dots \ n_{x,N-1}(k)]^T, \quad (2.47)$$

$$\mathbf{n}_y(k) = [n_{y,0}(k) \ n_{y,1}(k) \ \dots \ n_{y,N-1}(k)]^T, \quad (2.48)$$

$$\mathbf{s}(k) = [s_0(k) \ s_1(k) \ \dots \ s_{M-1}(k)]^T, \quad (2.49)$$



$$\mathbf{A}_X = \begin{bmatrix} a_0(\theta_0) & a_0(\theta_1) & \dots & a_0(\theta_{M-1}) \\ a_1(\theta_0) & a_1(\theta_1) & \dots & a_1(\theta_{M-1}) \\ \vdots & \vdots & \ddots & \vdots \\ a_{N-1}(\theta_0) & a_{N-1}(\theta_1) & \dots & a_{N-1}(\theta_{M-1}) \end{bmatrix}, \quad (2.50)$$

$$\mathbf{\Phi} = \begin{bmatrix} e^{\frac{j\omega\delta}{c}\text{sen}(\theta_0)} & 0 & \dots & 0 \\ 0 & e^{\frac{j\omega\delta}{c}\text{sen}(\theta_1)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{\frac{j\omega\delta}{c}\text{sen}(\theta_{M-1})} \end{bmatrix}. \quad (2.51)$$

Com isso, as equações (2.43) e (2.44) são descritas, no domínio do tempo discreto, respectivamente por

$$\mathbf{x}(k) = \mathbf{A}_X \mathbf{s}(k) + \mathbf{n}_x(k), \quad (2.52)$$

$$\mathbf{y}(k) = \mathbf{A}_X \mathbf{\Phi} \mathbf{s}(k) + \mathbf{n}_y(k). \quad (2.53)$$

A matriz  $\begin{bmatrix} \mathbf{A}_X \\ \mathbf{A}_X \mathbf{\Phi} \end{bmatrix}$  está relacionada aos ganhos das antenas de recepção em relação aos ângulos de chegada, é chamada de *array manifold*. O funcionamento do método ESPRIT requer que  $\begin{bmatrix} \mathbf{A}_X \\ \mathbf{A}_X \mathbf{\Phi} \end{bmatrix}$  tenha posto completo  $M$ .

A partir desta descrição de ambiente, a etapa SSI4 do processo de estimação de DoA é feita através da seguinte operação sobre as estimações  $\hat{\phi}_i$ , elementos na diagonal de  $\mathbf{\Phi}$ ,

$$\hat{\theta}_i = \arcsen \left( \frac{c}{j\omega\delta} \ln(\hat{\phi}_i) \right). \quad (2.54)$$

O vetor  $\boldsymbol{\theta}$  é formado colocando-se as estimações  $\hat{\theta}_i$  em suas linhas.

### 2.3.3 Algoritmo ESPRIT para Estimação de DoA

Considera-se uma modelagem alternativa, mais compacta do problema [14], que permite escrever

$$\mathbf{z}(k) = \mathbf{A} \mathbf{s}(k) + \mathbf{n}_z(k), \quad (2.55)$$

para  $\mathbf{z}(k)$  de comprimento  $2N$ , onde

$$\mathbf{z}(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix}, \quad (2.56)$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_X \\ \mathbf{A}_X \Phi \end{bmatrix}, \quad (2.57)$$

$$\mathbf{n}_z(k) = \begin{bmatrix} \mathbf{n}_x(k) \\ \mathbf{n}_y(k) \end{bmatrix}. \quad (2.58)$$

Considerando-se

$$\mathbf{R}_s = E[\mathbf{s}(k)\mathbf{s}^H(k)], \quad (2.59)$$

$$\mathbf{R}_z = E[\mathbf{z}(k)\mathbf{z}^H(k)], \quad (2.60)$$

$$\sigma_N^2 \mathbf{R}_N = E[\mathbf{n}_z(k)\mathbf{n}_z^H(k)], \quad (2.61)$$

o modelo de correlação para a equação (2.55) é expresso por

$$\mathbf{R}_z = \mathbf{A}\mathbf{R}_s\mathbf{A}^H + \sigma_N^2 \Sigma_N. \quad (2.62)$$

São realizadas então as duas primeiras etapas de estimação SSI, de acordo com as definições da subseção 2.3.1,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}(0) & \mathbf{z}(1) & \dots & \mathbf{z}(K-1) \end{bmatrix}, \quad (2.63)$$

$$[\mathbf{U}_s, \Lambda_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_z, \mathbf{R}_n, \text{SSI1}(\mathbf{Z})), \quad (2.64)$$

onde  $\mathbf{Z}$  se refere a uma matriz contendo em suas colunas todos os *snapshots* transmitidos.

Por causa da restrição de funcionamento de posto completo, a matriz  $\mathbf{A}$  deve possuir posto completo por colunas. Então, existe uma única matriz  $\mathbf{T}$ , de dimensões  $M \times M$ , com posto completo, tal que

$$\mathbf{U}_s = \mathbf{A}\mathbf{T}. \quad (2.65)$$

Particionando-se  $\mathbf{U}_s$  da forma

$$\mathbf{U}_s = \begin{bmatrix} \mathbf{U}_x \\ \mathbf{U}_y \end{bmatrix}, \quad (2.66)$$

têm-se

$$\mathbf{U}_x = \mathbf{A}_X \mathbf{T}, \quad (2.67)$$

$$\mathbf{U}_y = \mathbf{A}_X \Phi \mathbf{T}. \quad (2.68)$$

A equação de invariância do sistema é dada por

$$\mathbf{U}_x \Psi = \mathbf{U}_y. \quad (2.69)$$

A fim de resolver a equação de invariância, é definido um único par de matrizes  $\{\mathbf{F}_x, \mathbf{F}_y\}$  de posto completo tal que [14]

$$\mathbf{U}_x \mathbf{F}_x + \mathbf{U}_y \mathbf{F}_y = \mathbf{0}. \quad (2.70)$$

Define-se então uma nova matriz auxiliar  $\Psi$  tal que

$$\Psi = -\mathbf{F}_x \mathbf{F}_y^{-1}. \quad (2.71)$$

A partir das equações (2.67)–(2.71), tem-se que

$$\mathbf{A}_X \mathbf{T} \Psi = \mathbf{A}_X \Phi \mathbf{T}. \quad (2.72)$$

A matriz  $\Phi$  que contém informações sobre os ângulos de chegada é então estimada a partir de uma EVD sobre  $\Psi$ , ou seja,

$$\hat{\Phi} = \mathbf{T} \Psi \mathbf{T}^{-1}, \quad (2.73)$$

lembrando que  $\Phi$  é diagonal.

Normalmente não se tem acesso às médias estatísticas que dá origem à matriz de covariância  $\mathbf{R}_z$ , utilizando-se então uma estimação  $\hat{\mathbf{R}}_z$ . Com isso, a equação (2.65) pode corresponder a um sistema denominado *indeterminado* na álgebra linear, isto é, pode não existir uma matriz  $\mathbf{T}$  que satisfaça a equação (2.65) quando  $\hat{\mathbf{R}}_z$  não corresponder exatamente ao valor de  $\mathbf{R}_z$ . Obtém-se então uma estimação TLS para a matriz  $\Psi$  de acordo com a equação (2.38), ou seja,

$$[\hat{\Psi}] = \text{TLS}(\mathbf{U}_x, \mathbf{U}_y), \quad (2.74)$$

ou seja, para um sistema indeterminado, é possível ser feita uma estimação TLS. A matriz de parâmetros  $\Phi$  é estimada a partir de  $\Psi$  de acordo com a equação (2.73).

Por fim, é aplicada a função SSI4 para o ambiente de DoA, como definido na equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\Phi}). \quad (2.75)$$

Na Figura 2.6 está representado o processo de estimação realizado pelo ESPRIT.

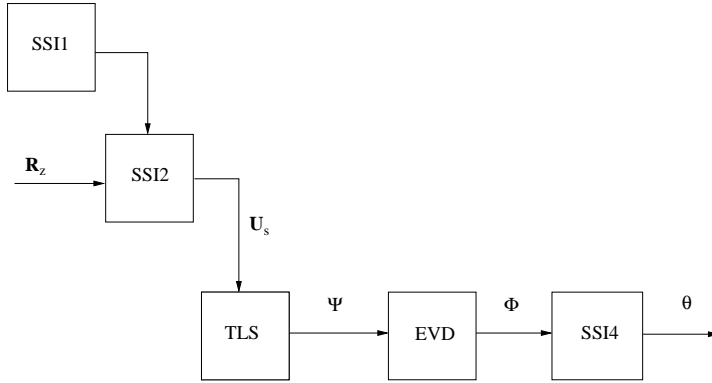


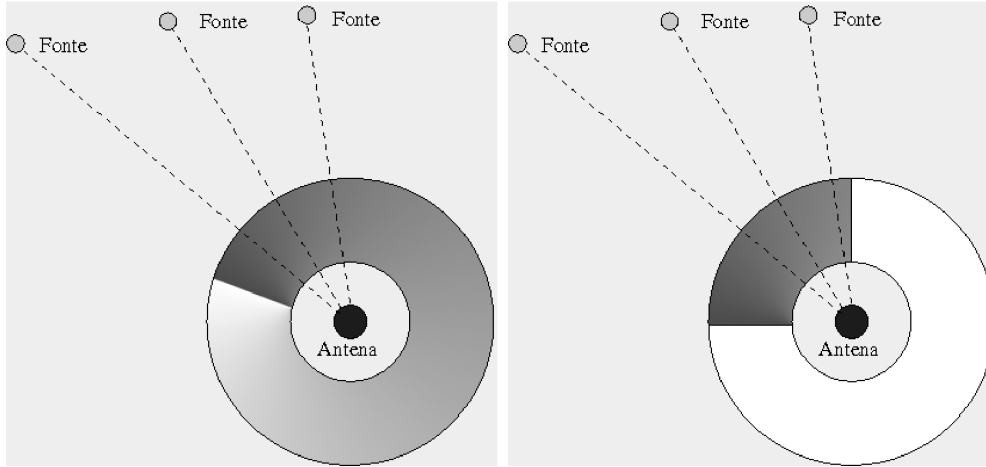
Figura 2.6: Diagrama de blocos representando o processo de estimação do algoritmo ESPRIT.

### 2.3.4 Processamento no *Beamspace* DFT

Na Subseção 2.3.3, os sinais efetivamente processados estão no próprio domínio de aquisição, cujo espaço correspondente de sinais processados é frequentemente conhecido como *elementspace* [24]. No entanto, o processamento pode ser realizado em um outro domínio, a fim de diminuir as dimensões das estruturas tratadas. O domínio escolhido é tal que corresponda a algum *beampattern* predefinido que mantenha as características do sinal desejado.

No caso de um *array* de recepção uniformemente espaçado (ULA - *uniformly linear array*), o *beampattern* e os coeficientes podem estar relacionados para um *grid* de “frequências” uniformemente espaçado. A seguir será descrito um algoritmo para estimação de DoA baseado em projeções no *beamspace* com matrizes DFT, de acordo com [25], que descreve um método ESPRIT trabalhando num *beamspace* gerado por matrizes DFT.

Ao se utilizar um *beamspace* DFT, pode-se interpretar os novos sinais gerados como correspondentes aos sinais originais no domínio da frequência. Deseja-se trabalhar com um *beamspace* onde estruturas possuam posto reduzido, ou seja, utilizando-se matrizes que projetam os dados originais em vetores com dimensão menor, de modo a serem obtidos algoritmos com menor complexidade computacional. Isso é realizado ao se gerar uma matriz de *beamspace* com a eliminação de algumas das colunas da matriz DFT, o que pode ser interpretado como sendo equivalente a uma varredura num espaço de DoAs com dimensões reduzidas. Essa interpretação é



(a) *BeamSpace* DFT com varredura completa.

(b) *BeamSpace* DFT com varredura reduzida.

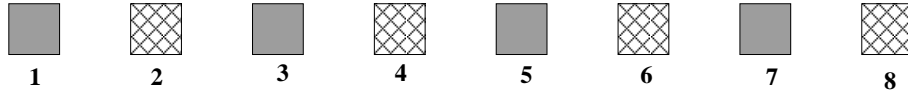
Figura 2.7: Interpretação da projeção no *BeamSpace* DFT vista como varredura num espaço de ângulos.

representada na Figura 2.7. Como a varredura é realizada em uma região reduzida, deve-se ter um conhecimento prévio de em qual região na qual se encontra a fonte desejada.

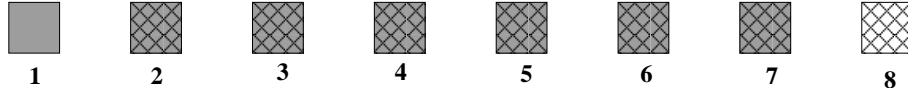
Caso não se tenha conhecimento prévio sobre qual a região de busca na qual se localiza a fonte, podem-se fazer buscas em regiões distintas. Como o processamento das buscas em regiões distintas pode ser realizado de maneira independente, uma redução de complexidade computacional em comparação com o processamento do *elementspace* pode ser obtido em uma máquina que permita paralelismo no processamento.

***BeamSpace* ESPRIT** O artigo [25] apresenta uma versão *beamSpace* DFT para o algoritmo ESPRIT. O algoritmo ESPRIT original, como visto na Subseção 2.3.3, impõe algumas restrições à geometria das antenas do *array* de recepção, de modo a garantir a existência de uma redundância na representação dos dados recebidos por dois *sub-arrays*. Como os dados recebidos são agora processados em um *beamSpace* DFT, novas restrições devem ser impostas à geometria do *array* a fim de ainda se garantir a redundância.

A utilização de um *beamSpace* DFT está ligada à formação de padrões de feixes



(a) *Elementspace* ESPRIT.



(b) *Beam-space* ESPRIT.

Figura 2.8: Escolhas de *sub-arrays* de recepção no *Elementspace* ESPRIT e no *Beam-space* ESPRIT. No caso *elementspace*, antenas  $(i, i + 1)$ ,  $i$  ímpar formam *doublets* com um *sub-array* de antenas com padrão cinza e outro representado com um padrão quadriculado. No caso *beam-space*, as antenas com padrão cinza quadriculado pertencem a ambos *sub-arrays*, com uma sobreposição de elementos entre os *doublets*.

(*beams*) que correspondem à DFT do filtro de recepção quando o *array* de recepção é um ULA [24]. Nesse caso, o uso de um processamento *beam-space* leva à síntese adequada do espectro de chegada somente em um ULA. Por isso, uma restrição geométrica adequada para um *beam-space* DFT é que o *array* de recepção seja um ULA.

Outra restrição diz respeito à escolha dos *sub-arrays* de recepção. Enquanto na abordagem ESPRIT tradicional, não há necessidade de sobreposição de *sub-arrays*, o algoritmo *Beam-space* ESPRIT requer sobreposição de *sub-arrays*, o que aumenta a complexidade computacional do algoritmo para uma mesma quantidade de antenas. Na Figura 2.8 estão representadas as escolhas de *sub-arrays* usadas na descrição *Elementspace* ESPRIT e no novo *Beam-space* ESPRIT. Na Figura 2.8a, as antenas de número par (indicados pela cor cinza) formam um dos *sub-arrays* enquanto as de número ímpar formam o outro, indicado pelo padrão quadriculado. Já na Figura 2.8b, as antenas numeradas de 2 a 7, preenchidas com os padrões quadriculado e cinza, pertencem a ambos *sub-arrays*.

A modelagem utilizada basicamente segue aquela apresentada na subseção 2.3.3.

Usa-se uma modelagem modificada do sistema em relação às equações (2.52) e (2.53)

$$z_i(t) = \sum_{m=0}^{M-1} s_m(t) a_i(\theta_m) + n_{z,i}(t), \quad (2.76)$$

A modelagem da equação (2.76) está relacionada com a equação vetorial (2.55), apresentada para o caso *elementspace*.

Amostrando-se o sinal recebido na mesma taxa que os símbolos transmitidos, definem-se alguns vetores e matrizes auxiliares no tempo discreto  $k$ ,

$$\mathbf{z}(k) = \begin{bmatrix} z_0(k) & z_1(k) & \dots & z_{N-1}(k) \end{bmatrix}^T, \quad (2.77)$$

$$\mathbf{n}_z(k) = \begin{bmatrix} n_{z,0}(k) & n_{z,1}(k) & \dots & n_{z,N-1}(k) \end{bmatrix}^T, \quad (2.78)$$

Com isso, a modelagem discreta do sistema passa a ser feita através de vetores e matrizes por

$$\mathbf{z}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{n}_z(k). \quad (2.79)$$

São também definidas matrizes de seleção  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  tais que

$$\mathbf{J}_X = \begin{bmatrix} \mathbf{I}_K & \mathbf{0}_{\{K \times (N-K)\}} \end{bmatrix}, \quad (2.80)$$

$$\mathbf{J}_Y = \begin{bmatrix} \mathbf{0}_{\{K \times (N-K)\}} & \mathbf{I}_K \end{bmatrix}, \quad (2.81)$$

ou seja,  $\mathbf{J}_X^T \mathbf{J}_X = \mathbf{J}_Y^T \mathbf{J}_Y = \mathbf{I}_K$ , onde  $\mathbf{I}_K$  representa uma matriz identidade de dimensões  $K \times K$  e  $\mathbf{0}_{\{K \times (N-K)\}}$  representa uma matriz contendo apenas elementos nulos, de dimensões  $K \times (N - K)$ . A variável  $K$  é um parâmetro de projeto ligado ao grau de sobreposição entre os *sub-arrays*.

No contexto de um ULA na recepção, há uma distância constante entre as antenas. Seja  $\delta$  o deslocamento entre antenas receptoras adjacentes. Então  $\mathbf{A}$  assume uma estrutura do tipo Vandermonde [24], que é um caso particular do visto para o *elementspace* na seção anterior. Com isso,  $\mathbf{A}$  atende automaticamente às restrições de invariância à rotação, como definido na Subseção 2.3.3, mais uma vez dando origem a uma equação de invariância,

$$\mathbf{J}_X \mathbf{A} \Phi = \mathbf{J}_Y \mathbf{A}, \quad (2.82)$$

lembrando que a matriz  $\mathbf{A}$  nesta subseção tem dimensões  $N \times M$ .

Na abordagem *beamspace*, o vetor  $\mathbf{z}(k)$  é processado por transformadas ortogonais  $\mathbf{T}_i^H$ , com dimensões  $N \times L$ , onde  $L < N$ . Considerando-se o componente AWGN descorrelacionado com as fontes, o modelo de covariância para o sistema é expressado por

$$\mathbf{R}_z = \mathbf{T}_i^H \mathbf{A} \mathbf{R}_s \mathbf{A}^H \mathbf{T}_i + \sigma_N^2 \mathbf{I}, \quad (2.83)$$

onde  $\mathbf{R}_z$  e  $\mathbf{R}_s$  representam as matrizes de autocovariância dos sinais recebido e transmitido, respectivamente.

Na modelagem apresentada para o *Elementspace* ESPRIT, a matriz  $\mathbf{A}$  contém uma redundância em sua estrutura que dá origem à propriedade de invariância à rotação. Com o uso de matrizes de transformação para o *beamspace*, a ortogonalidade não é mais garantida. Para o funcionamento adequado do *Beamspace* ESPRIT,  $\mathbf{T}_i^H \mathbf{A}$  deve manter as mesmas propriedades de invariância à rotação que a matriz  $\mathbf{A}$  da versão *elementspace* possui.

Considere que a transformação  $\mathbf{T}_i$  aplicada a  $\mathbf{z}(k)$  satisfaça

$$\mathbf{J}_X \mathbf{T}_i = \mathbf{J}_Y \mathbf{T}_i \mathbf{F}, \quad (2.84)$$

para uma matriz  $\mathbf{F}$  qualquer, com posto completo. Considere ainda que  $\mathbf{t}_{i,j}$  a  $j$ -ésima coluna de  $\mathbf{T}_i$ , para  $0 \leq i \leq N - 1$ . Caso exista uma matriz  $\mathbf{Q}$ , tal que

$$\begin{cases} \mathbf{Q} \mathbf{F}^H \mathbf{t}_{i,j} = 0, & 0 \leq j < N - K, \\ \mathbf{Q} \mathbf{t}_{i,j} = 0, & K \leq j < N, \end{cases} \quad (2.85)$$

então, a partir das equações (2.82) e (2.84), conclui-se que

$$\begin{aligned} \mathbf{Q} \mathbf{T}_i^H \mathbf{A} &= \mathbf{Q} \mathbf{T}_i^H \mathbf{J}_X^H \mathbf{J}_X \mathbf{A} \\ &= \mathbf{Q} \mathbf{T}_i^H \mathbf{J}_X^H \mathbf{J}_Y \mathbf{A} \mathbf{F}^H \\ &= \mathbf{Q} \mathbf{F}^H \mathbf{T}_i^H \mathbf{J}_Y^H \mathbf{J}_Y \mathbf{A} \mathbf{F}^H \\ &= \mathbf{Q} \mathbf{F}^H \mathbf{T}_i^H \mathbf{A} \mathbf{\Psi}^H. \end{aligned} \quad (2.86)$$

Mais uma vez, são realizadas as duas primeiras etapas do sistema SSI, como definido na Subseção 2.3.1,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}(0) & \mathbf{z}(1) & \dots & \mathbf{z}(K-1) \end{bmatrix}, \quad (2.87)$$

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_z, \text{SSI1}(\mathbf{Z})). \quad (2.88)$$



A partir da equação (2.86), chega-se a uma nova expressão para a equação de invariância

$$\mathbf{Q}\mathbf{T}_i^H\mathbf{U}_s = \mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{U}_s\mathbf{\Phi}^H. \quad (2.89)$$

Do mesmo modo que o *Elementspace* ESPRIT, na versão *beamspace*  $\hat{\Psi}$  pode ser estimada por TLS, através da equação (2.38),

$$[\hat{\Psi}] = \text{TLS}(\mathbf{U}_X, \mathbf{U}_Y), \quad (2.90)$$

onde, mais uma vez  $\mathbf{U}_X = \mathbf{J}_X\mathbf{U}_s$  e  $\mathbf{U}_Y = \mathbf{J}_Y\mathbf{U}_s$ . Descobre-se então  $\mathbf{\Phi}$  através de uma EVD sobre  $\Psi$

$$\Psi = \mathbf{E}_P\mathbf{\Phi}\mathbf{E}_P^{-1}, \quad (2.91)$$

onde  $\mathbf{E}_P$  é uma matriz que representa a relação entre o *array manifold*  $\mathbf{A}$  e o subespaço dos sinais gerado por  $\mathbf{U}_X$ . Por fim, aplica-se a função SSI4 sobre a estimação  $\hat{\mathbf{\Phi}}$  de acordo com a equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (2.92)$$

### 2.3.5 *Unitary* ESPRIT

Nas subseções anteriores, o processamento do algoritmo ESPRIT é feito usando-se operações e estruturas complexas. Como as operações sobre estruturas complexas possuem um maior número de operações que as equivalentes reais, pode-se afirmar que boa parte da complexidade computacional do sistema se deve à utilização de estruturas complexas. Em [37] é proposta uma alternativa ao ESPRIT em que a maior parte de seu processamento ocorre apenas com estruturas reais. No entanto, tal algoritmo, que recebeu a denominação *Unitary* ESPRIT, impõe algumas restrições extras sobre a geometria do *array* de recepção. Além disso, algumas estruturas possuem suas dimensões aumentadas ao se passar para o domínio de operações reais. Antenas com as mesmas características de recepção devem estar equidistantes do centro geométrico do *array*, chamado centróide do *array*, como mostrado na Figura 2.9. Na Seção 3.7 é feita uma comparação da complexidade computacional entre operações reais e complexas.

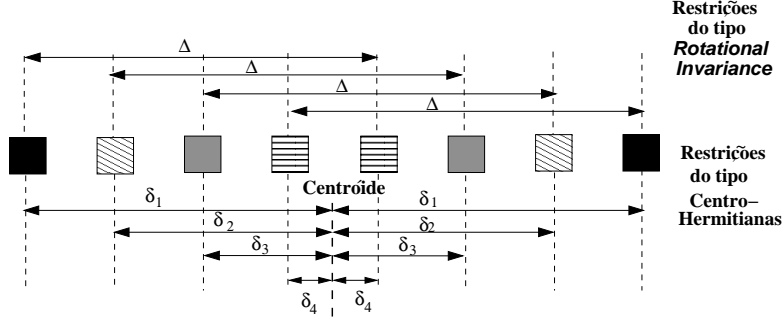


Figura 2.9: Antenas de recepção com as restrições geométricas referentes à invariância à rotação e as novas restrições para *arrays* centro-hermitianos. Antenas representadas com o mesmo padrão de preenchimento possuem as mesmas características de recepção.

**Estruturas Utilizadas:** Considere mais uma vez a modelagem apresentada na equação (2.55), repetida a seguir, por conveniência,

$$\mathbf{z}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{n}_z(k).$$

Mais uma vez, definem-se matrizes de seleção  $\mathbf{J}_X$  e  $\mathbf{J}_Y$ , tais que  $\mathbf{x}(k) = \mathbf{J}_X\mathbf{z}(k)$  e  $\mathbf{y}(k) = \mathbf{J}_Y\mathbf{z}(k)$ , onde  $\mathbf{x}(k)$  e  $\mathbf{y}(k)$  estão definidos como nas equações (2.52) e (2.53). Definem-se então  $\mathbf{A}_X$  e  $\mathbf{A}_Y$  tais que

$$\mathbf{A}_X = \mathbf{J}_X\mathbf{A}, \quad (2.93)$$

$$\mathbf{A}_Y = \mathbf{J}_Y\mathbf{A}. \quad (2.94)$$

Como já mostrado na Subseção 2.3.3,

$$\mathbf{A}_Y = \mathbf{A}_X\mathbf{\Phi}, \quad (2.95)$$

que representa a equação de invariância do ESPRIT original. Define-se então a matriz de permutação  $\mathbf{\Pi}_i$  de dimensões  $i \times i$ , que possui 1 em sua antidiagonal e zero nos demais elementos. Uma matriz  $\mathbf{A}$  é dita centro-hermitiana (CH) [37] se for satisfeita a seguinte equação,

$$\mathbf{\Pi}_{2N}\mathbf{A}^*\mathbf{\Pi}_M = \mathbf{A}, \quad (2.96)$$

onde o operador  $*$  retorna o complexo conjugado de cada elemento da matriz. O nome CH vem do fato de esse tipo de matriz ter como entradas o complexo conjugado das entradas de uma matriz cujas entradas estão simetricamente trocadas

na vertical. De maneira equivalente, a matriz é dita hermitiana se apresentar como entradas o complexo conjugado das entradas de uma matriz com entradas trocadas simetricamente trocadas em relação à sua diagonal. A equação (2.99) caracteriza  $\bar{\mathbf{A}}$  como uma matriz CH, ou seja, se cada coluna tiver seus elementos colocados numa ordem invertida (um *flip* na vertical) a nova matriz deve ter como elementos o complexo conjugado de cada elemento da matriz original. Por exemplo, considere  $\mathbf{a}_i$  como a  $i$ -ésima coluna de  $\mathbf{A}$ . Considerando

$$\mathbf{a}_i = \begin{bmatrix} a_{i0} \\ a_{i1} \\ \vdots \\ a_{i(M-1)} \end{bmatrix}, \quad (2.97)$$

então

$$\begin{bmatrix} a_{i(M-1)} \\ a_{i(M-2)} \\ \vdots \\ a_{i0} \end{bmatrix} = \begin{bmatrix} a_{i0}^* \\ a_{i1}^* \\ \vdots \\ a_{i(M-1)}^* \end{bmatrix} \quad (2.98)$$

Já para um *array manifold* ser CH, as restrições apresentadas na Figura 2.9 se traduzem matematicamente em critérios matemáticos um pouco mais relaxados que os representados pela equação (2.96) para uma matriz CH,

$$\mathbf{\Pi}_{2N} \mathbf{A}^* = \mathbf{A} \mathbf{\Lambda}_{CH}, \quad (2.99)$$

tal que  $\mathbf{\Lambda}_{CH}$  seja uma matriz diagonal e unitária.

Considere uma matriz de rotação  $\mathbf{Q}_a$ , com  $a$  par, definida por

$$\mathbf{Q}_a = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{a/2 \times a/2} & j\mathbf{I}_{a/2 \times a/2} \\ \mathbf{\Pi}_{a/2} & -j\mathbf{\Pi}_{a/2} \end{bmatrix}. \quad (2.100)$$

Com o auxílio de  $\mathbf{Q}_a$  é definida uma transformação  $\psi(\mathbf{M})$  tal que

$$\psi(\mathbf{M}) = \mathbf{Q}_a^H \mathbf{M} \mathbf{Q}_a. \quad (2.101)$$

É mostrado no artigo [37] que, caso o argumento da transformação  $\psi(\cdot)$  seja uma matriz CH, então a transformação  $\psi$  gera uma matriz apenas com elementos reais. Essa propriedade é a chave para a definição do algoritmo ESPRIT apenas com operações reais.

As deduções a seguir pressupõem  $M$ , a quantidade de antenas transmissoras, como sendo um número par. Caso fosse um número ímpar, a dedução seria semelhante, mudando-se apenas a definição de  $\mathbf{Q}_a$ , dada originalmente pela equação (2.100) para [1]

$$\mathbf{Q}_a = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{a/2 \times a/2} & \mathbf{0} & j\mathbf{I}_{a/2 \times a/2} \\ \mathbf{0}^T & \sqrt{2} & \mathbf{0}^T \\ \mathbf{\Pi}_{a/2} & \mathbf{0} & -j\mathbf{\Pi}_{a/2} \end{bmatrix}. \quad (2.102)$$

Voltando-se à matriz  $\mathbf{Q}_a$  conforme definida na equação (2.100), a matriz  $\mathbf{U}_M$  contendo em suas colunas os autovetores de uma matriz  $\mathbf{M}$  e a matriz  $\mathbf{U}_\psi(\mathbf{M})$  de autovetores de sua transformação  $\psi(\mathbf{M})$  estão relacionadas por

$$\mathbf{U}_M = \mathbf{Q}_a \mathbf{U}_\psi(\mathbf{M}). \quad (2.103)$$

Para qualquer matriz  $\mathbf{M}$ , é mostrado em [37] que uma transformação  $\mathcal{E}(\cdot)$ , definida por

$$\mathcal{E}(\mathbf{M}) = \begin{bmatrix} \mathbf{M} & \mathbf{\Pi}_a \mathbf{M}^* \mathbf{\Pi}_a \end{bmatrix}, \quad (2.104)$$

gera sempre uma matriz CH.

Usando-se então as equações (2.101) e (2.104), conclui-se que, a partir de uma matriz  $\mathbf{M}$  qualquer, a transformação

$$\mathcal{T}(\mathbf{M}) = \psi(\mathcal{E}(\mathbf{M})) \quad (2.105)$$

gera uma matriz apenas com elementos reais.

Considerando as partições  $\mathbf{M}_X$  e  $\mathbf{M}_Y$  de  $\mathbf{M}$ , ou seja,

$$\mathbf{M}_X = \mathbf{J}_X \mathbf{M}, \quad (2.106)$$

$$\mathbf{M}_Y = \mathbf{J}_Y \mathbf{M}, \quad (2.107)$$

então, é mostrado em [37] que  $\mathcal{T}(\mathbf{M})$  está relacionado com  $\mathbf{M}_X$  e  $\mathbf{M}_Y$  através da seguinte expressão

$$\mathcal{T}(\mathbf{M}) = \begin{bmatrix} \Re(\mathbf{M}_X + \mathbf{\Pi}_N \mathbf{M}_Y^*) & -\Im(\mathbf{M}_X - \mathbf{\Pi}_N \mathbf{M}_Y^*) \\ \Im(\mathbf{M}_X + \mathbf{\Pi}_N \mathbf{M}_Y^*) & \Re(\mathbf{M}_X - \mathbf{\Pi}_N \mathbf{M}_Y^*) \end{bmatrix}, \quad (2.108)$$

onde o operador  $\Re(\cdot)$  retorna a parte real do seu argumento enquanto  $\Im(\cdot)$  retorna sua parte imaginária. Verifica-se a partir da equação (2.108) que a computação

de  $\mathcal{T}(\mathbf{M})$  apresenta uma complexidade pequena, envolvendo basicamente adições, subtrações e permutações de matrizes. Esse pequeno *overhead* de complexidade e o ganho proporcionado pelo algoritmo no novo domínio proporcionam a redução total de complexidade computacional do *Unitary* ESPRIT em relação ao original.

Considere uma matriz  $\mathbf{Z}(i)$  contendo  $K$  *snapshots* de  $\mathbf{z}(k)$  em suas colunas, ou seja,

$$\mathbf{Z}(i) = \begin{bmatrix} \mathbf{z}(i) & \mathbf{z}(i-1) & \dots & \mathbf{z}(i-K) \end{bmatrix}, \quad (2.109)$$

A partir das propriedades já definidas para a transformação  $\mathcal{T}(\cdot)$ , a matriz  $\mathcal{T}(\mathbf{Z}(i))$  representa uma estrutura apenas com elementos reais. Essa é a estrutura básica a partir da qual é deduzido o algoritmo.

A seguir é então definida uma estrutura  $\mathbf{Z}_{FB}(i)$  do tipo *forward-backward* (FB)

$$\mathbf{Z}_{FB}(i) = \begin{bmatrix} \mathbf{Z}(i) & \mathbf{\Pi}_{2N} \mathbf{Z}^*(i) \end{bmatrix}. \quad (2.110)$$

A matriz  $\mathbf{Z}_{FB}(i)$  pode ser decomposta da seguinte forma

$$\mathbf{Z}_{FB}(i) = \begin{bmatrix} \mathbf{X} & \mathbf{\Pi}_M \mathbf{Y}^* \\ \mathbf{Y} & \mathbf{\Pi}_M \mathbf{X}^* \end{bmatrix}, \quad (2.111)$$

o que, por sua vez, leva à fatoração [37]

$$\mathbf{Z}_{FB}(i) = \begin{bmatrix} \mathbf{A}_X \\ \mathbf{A}_X \mathbf{\Phi} \end{bmatrix} \begin{bmatrix} \mathbf{S} & \mathbf{\Phi}^{-1} \mathbf{S}^* \end{bmatrix} \quad (2.112)$$

**Nova Equação de Invariância:** É definida então a covariância  $\mathbf{R}$  dos dados reais representados por  $\mathcal{T}(\mathbf{Z}(i))$  num intervalo de  $K$  amostras, ou seja,

$$\mathbf{R} = \mathcal{T}(\mathbf{Z}(i)) \mathcal{T}^H(\mathbf{Z}(i)). \quad (2.113)$$

A covariância  $\mathbf{R}$ , no entanto, também pode ser expressa a partir da matriz FB de dados  $\mathbf{Z}_{FB}$

$$\mathbf{R} = \mathbf{Q}_{2N}^H \mathbf{Z}_{FB} \mathbf{Z}_{FB}^H \mathbf{Q}_{2N} = \psi(\mathbf{Z}_{FB} \mathbf{Z}_{FB}^H). \quad (2.114)$$

Como as matrizes  $\mathbf{R}$  e  $\mathbf{Z}_{FB} \mathbf{Z}_{FB}^H$  estão relacionadas pela transformação  $\psi(\cdot)$ , então suas respectivas matrizes de autovetores estão relacionadas de acordo com a equação (2.103), isto é,

$$\mathbf{U}_R = \mathbf{Q}_{2N} \mathbf{U}_{ZZ}. \quad (2.115)$$

Particiona-se a matriz  $\mathbf{U}_{ZZ}$  de modo que

$$\mathbf{C}_X = \mathbf{J}_X \mathbf{U}_{ZZ}, \quad (2.116)$$

$$\mathbf{C}_Y = \mathbf{J}_Y \mathbf{U}_{ZZ}. \quad (2.117)$$

As matrizes  $\mathbf{C}_X$  e  $\mathbf{C}_Y$  representam partes da estrutura de  $\mathbf{U}_{ZZ}$ . A matriz  $\mathbf{U}_{ZZ}$  gera o mesmo subespaço dos sinais gerado pela matriz  $\mathbf{Z}_{FB}$ . Como é mostrado na equação (2.112),  $\mathbf{Z}_{FB}$  possui uma regularidade em sua estrutura, gerando uma invariância à rotação semelhante à gerada pela estrutura  $\mathbf{Z}$ . Com isso, assim como no caso do ESPRIT original, apresentado na Subseção 2.3.3, a matriz  $\mathbf{U}_{ZZ}$  de autovetores dos dados, apresenta uma redundância na sua estrutura. Essa regularidade na estrutura pode ser expressa pela seguinte equação de invariância [37]

$$\mathbf{C}_X \boldsymbol{\Psi} = \mathbf{C}_Y. \quad (2.118)$$

Essas estruturas já são matrizes com elementos reais. A fim de que se possa deduzir um algoritmo com equivalência entre soluções de estimações TLS, é necessário que a estimação da matriz  $\mathbf{U}_{ZZ}$ , e, conseqüentemente, de  $\mathbf{C}_X$  e  $\mathbf{C}_Y$ , seja feita apenas com estruturas reais. Como apresentado na equação (2.108), a transformação  $\mathcal{T}(\cdot)$  pode ser utilizada para se trabalhar apenas com elementos reais. Procura-se também um método de estimação do subespaço dos sinais a partir de  $\mathcal{T}(\mathbf{Z}(k))$ .

**Solução TLS com Elementos Reais:** Define-se a estrutura auxiliar

$$\mathbf{Z}_{FB,CH} = \mathbf{Z}_{FB} \begin{bmatrix} \mathbf{I}_{N \times N} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Pi}_N \end{bmatrix} = \begin{bmatrix} \mathbf{Z}(i) & \boldsymbol{\Pi}_{2N} \mathbf{Z}^*(i) \boldsymbol{\Pi}_M \end{bmatrix}. \quad (2.119)$$

A matriz  $\mathbf{Z}_{FB,CH}$  representa uma transformação do tipo  $\mathcal{E}(\cdot)$ , ou seja, é uma matriz CH. Além disso, verifica-se a relação  $\mathbf{Z}_{FB,CH} = \mathcal{E}(\mathbf{Z}(i))$ . Utilizando-se a equação (2.105), pode-se dizer que

$$\psi(\mathbf{Z}_{FB,CH}) = \mathcal{T}(\mathbf{Z}(i)), \quad (2.120)$$

o que possibilita a utilização de estruturas apenas com elementos reais na estimação do subespaço dos sinais.

Em ambientes com presença de AWGN, a equação de invariância expressa em (2.118) é considerada indeterminada, isto é, não apresenta solução através da in-

versão direta da matriz  $\mathbf{C}_X$ . O problema pode ser resolvido através de uma abordagem TLS, como mostrado na Subseção 2.3.3. A solução TLS pode ser obtida através dos autovetores da matriz

$$\mathbf{C}_{TLS} = \begin{bmatrix} \mathbf{C}_X^H \\ \mathbf{C}_Y^H \end{bmatrix} \begin{bmatrix} \mathbf{C}_X & \mathbf{C}_Y \end{bmatrix}. \quad (2.121)$$

O cálculo de  $\mathbf{C}_{TLS}$  exige que sejam feitas operações com números complexos.

Deseja-se então utilizar as relações de autovalores já definidas nesta seção de modo a se ter operações apenas com elementos reais. Então, a partir da relação de autovetores expressa pela equação (2.103), pode-se dizer que

$$\mathbf{U}_{ZZ} = \begin{bmatrix} \mathbf{C}_X \\ \mathbf{C}_Y \end{bmatrix} = \begin{bmatrix} \mathbf{J}_X \\ \mathbf{J}_Y \end{bmatrix} \mathbf{Q}_M \mathbf{U}_{Z,FB,CH}. \quad (2.122)$$

A partir da equação (2.122) tem-se que

$$\mathbf{C}_X = \mathbf{J}_X \mathbf{Q}_M \mathbf{U}_{Z,FB,CH}, \quad (2.123)$$

onde  $\mathbf{U}_{Z,FB,CH}$  representa a matriz com autovetores de  $\mathbf{U}_{FB,CH}$ . Com isso,

$$\mathbf{\Pi}_M \mathbf{C}_X^* = \mathbf{\Pi}_M \mathbf{J}_X \mathbf{Q}_{2N}^* \mathbf{U}_{Z,FB,CH}. \quad (2.124)$$

Como  $\mathbf{J}_Y = \mathbf{\Pi}_M \mathbf{J}_X \mathbf{\Pi}_{2N}$ , então

$$\mathbf{\Pi}_M \mathbf{C}_X^* = \mathbf{\Pi}_M \mathbf{\Pi}_M \mathbf{J}_Y \mathbf{\Pi}_{2N} \mathbf{Q}_{2N}^* \mathbf{U}_{Z,FB,CH} = \mathbf{J}_Y \mathbf{Q}_{2N} \mathbf{U}_{Z,FB,CH} = \mathbf{C}_Y. \quad (2.125)$$

A partir da equação (2.125), verifica-se a seguinte equivalência

$$\begin{bmatrix} \mathbf{C}_X & \mathbf{C}_Y \end{bmatrix} = \begin{bmatrix} \mathbf{C}_X & \mathbf{\Pi}_M \mathbf{C}_X^* \end{bmatrix}. \quad (2.126)$$

Considera-se a matriz auxiliar

$$\mathbf{C}_{CH} = \begin{bmatrix} \mathbf{C}_X & \mathbf{\Pi}_M \mathbf{C}_X^* \end{bmatrix} \begin{bmatrix} \mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_K \end{bmatrix}. \quad (2.127)$$

Pode-se perceber a seguinte relação

$$\mathcal{E}(\mathbf{C}_{CH}) = \mathcal{T}(\mathbf{C}_X). \quad (2.128)$$

Considera-se então

$$\mathbf{P} = \mathcal{T}(\mathbf{C}_X^H \mathbf{C}_X) = \mathbf{Q}_{2K}^H \begin{bmatrix} \mathbf{I}_{K \times K} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi}_K \end{bmatrix} \mathbf{C}_{TLS} \begin{bmatrix} \mathbf{I}_{K \times K} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi}_K \end{bmatrix} \mathbf{Q}_{2K}. \quad (2.129)$$

Define-se  $\mathbf{U}_P$  como uma matriz com os autovetores de  $\mathbf{P}$  em suas colunas e também é definida  $\mathbf{U}_{CTLS}$  a matriz contendo os autovetores de  $\mathbf{C}_{TLS}$  em suas colunas. As matrizes  $\mathbf{U}_P$  e  $\mathbf{U}_{CTLS}$  se relacionam por

$$\mathbf{U}_{CTLS} = \begin{bmatrix} \mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi} \end{bmatrix} \mathbf{Q}_{2M} \mathbf{U}_P = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_N & j\mathbf{I}_N \\ \mathbf{I}_N & -j\mathbf{I}_N \end{bmatrix} \mathbf{U}_P \quad (2.130)$$

Particionando-se as matrizes  $\mathbf{U}_P$  e  $\mathbf{U}_{CTLS}$  em submatrizes de mesmas dimensões

$$\mathbf{U}_P = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix}, \quad (2.131)$$

$$\mathbf{U}_{CTLS} = \begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{U}_{21} & \mathbf{U}_{22} \end{bmatrix}. \quad (2.132)$$

As soluções TLS para cada um dos problemas podem ser representadas por

$$\mathbf{\Upsilon}_P = -\mathbf{W}_{12} \mathbf{W}_{22}^{-1}, \quad (2.133)$$

$$\mathbf{\Psi}_{CTLS} = -\mathbf{U}_{12} \mathbf{U}_{22}^{-1}. \quad (2.134)$$

A partir das equações (2.130), (2.133) e (2.134), pode-se reescrever a equação (2.134) como

$$\mathbf{\Psi}_{CTLS} = -(\mathbf{W}_{12} + j\mathbf{W}_{22})(\mathbf{W}_{12} - j\mathbf{W}_{22})^{-1}, \quad (2.135)$$

$$= -((-\mathbf{W}_{12} \mathbf{W}_{22}^{-1}) - j\mathbf{I}_M)((-\mathbf{W}_{12} \mathbf{W}_{22}^{-1}) + j\mathbf{I}_M)^{-1}, \quad (2.136)$$

$$= -(\mathbf{\Upsilon}_P - j\mathbf{I}_M)(\mathbf{\Upsilon}_P + j\mathbf{I}_M)^{-1}. \quad (2.137)$$

A equação (2.137) mostra que é possível se resolver uma equação de invariância com matrizes complexas a partir de um outro problema em que apenas estruturas reais são utilizadas.

Aplicando-se EVD sobre  $\mathbf{\Upsilon}_P$

$$\mathbf{\Upsilon}_P = \mathbf{T}_P \mathbf{\Omega} \mathbf{T}_P^{-1}, \quad (2.138)$$

é afirmado em [37] que a matriz  $\hat{\mathbf{\Phi}}$  de DoA estimados está relacionada com  $\mathbf{\Omega}$  por

$$\hat{\mathbf{\Phi}} = -(\mathbf{\Omega} - j\mathbf{I}_M)(\mathbf{\Omega} + j\mathbf{I}_M)^{-1}. \quad (2.139)$$

As equações (2.105), (2.137) e (2.139) juntas possibilitam o cálculo da solução TLS de maneira semelhante ao ESPRIT utilizando-se apenas estruturas com elementos reais, gerando o algoritmo denominado *Unitary ESPRIT*, como mostrado a seguir.



**Estimação** São primeiramente realizadas as duas primeiras etapas de um processo de estimação SSI, de acordo com o visto na Subseção 2.3.1,

$$[\mathbf{U}_{R_s}, \mathbf{\Lambda}_{R_s}^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}, \text{SSI1}(\mathcal{T}(\mathbf{C}_X))). \quad (2.140)$$

São usadas as matrizes de seleção  $\mathbf{K}_X$  e  $\mathbf{K}_Y$  sobre  $\mathbf{U}_{R_s}$ , onde

$$\mathbf{K}_X = \mathbf{Q}_N^H (\mathbf{J}_X + \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{2N}) \mathbf{Q}_{2N}, \quad (2.141)$$

$$\mathbf{K}_Y = \mathbf{Q}_N^H j(\mathbf{J}_X - \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{2N}) \mathbf{Q}_{2N}. \quad (2.142)$$

Ao se usarem essas matrizes, deseja-se obedecer a equação de invariância definida a seguir,

$$\mathbf{K}_X \mathbf{U}_{R_s} \mathbf{\Upsilon} = \mathbf{K}_Y \mathbf{U}_{R_s}. \quad (2.143)$$

É encontrada a solução TLS  $\mathbf{\Upsilon}_{\text{TLS}}$  para a equação (2.143), através da equação (2.38), ou seja,

$$[\mathbf{\Upsilon}_{\text{TLS}}] = \text{TLS}(\mathbf{K}_X \mathbf{U}_{R_s}, \mathbf{K}_Y \mathbf{U}_{R_s}). \quad (2.144)$$

A partir da equação (2.138) encontram-se os parâmetros  $\mathbf{\Omega}$ . Os parâmetros  $\mathbf{\Phi}$  são encontrados a partir da equação (2.139). Por fim, é realizada a operação SSI4 para o contexto de estimação de DoA, como definida na equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\phi}). \quad (2.145)$$

## 2.4 Métodos do tipo MP (*Matrix Pencil*)

O objetivo desta subseção é apresentar um método existente na literatura que estima DoA na recepção e que possui formulação com algumas similaridades com os algoritmos propostos a serem apresentados nos Capítulos 3 e 4. Tal método, conhecido como MP [18] é bem menos citado e utilizado que o ESPRIT. Sua comparação com os algoritmos propostos é então exposta de uma maneira resumida no Capítulo 3. Por isso, sua descrição será feita mais resumidamente que no caso dos algoritmos com base no ESPRIT.

Um sistema ESPRIT com base no uso de MPs usado na aplicação de *harmonic retrieval* é mostrado em [38]. O algoritmo trabalha com as mesmas estruturas a serem apresentadas no CB-DoA, porém o processamento é diferente, baseando-se

em um MP. Além disso, exige o uso de *arrays manifolds* com estrutura do tipo Vandermonde, semelhante ao de ULAs. A seguir será detalhado um outro método [18] com base em MP para estimação de DoA e *harmonic retrieval*.

Na modelagem do método MP apresentado em [18], considera-se que uma mistura de  $M$  senóides é recebida num *array* de  $N$  antenas, isto é,

$$x(k) = \sum_{t=1}^M |b_t| e^{((\alpha_t + j\omega_t)k + j\phi_t)} + n(k), \quad 0 \leq k \leq K - 1. \quad (2.146)$$

onde  $|b_t|$  representa a amplitude do sinal,  $\phi_t$  é a fase,  $\alpha_t$  é o fator de decaimento e  $\omega_t$  representa as frequências das senóides. Considerando a princípio que não haja AWGN no sistema, o sinal recebido pode ser dividido na parte de informação transmitida  $b_t$  e de desvanecimento  $z_t$  pela seguinte relação

$$x(k) = \sum_{t=1}^M b_t z_t, \quad (2.147)$$

onde

$$b_t = |b_t| e^{j\phi_t}, \quad (2.148)$$

$$z_t = e^{(\alpha_t + j\omega_t)k}. \quad (2.149)$$

O artigo [18] tem como objetivo a estimação dos parâmetros  $z_t$ . Para isso, consideram-se as seguintes estruturas auxiliares de dados [18]

$$\mathbf{X} = \begin{bmatrix} x_{L-1} \\ x_{L-2} \\ \vdots \\ x_0 \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} x_L \\ x_{L-1} \\ \vdots \\ x_1 \end{bmatrix}, \quad (2.150)$$

$$\mathbf{Z}_L = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_M \\ \vdots & \vdots & \ddots & \vdots \\ z_1^{N-L-1} & z_2^{N-L-1} & \dots & z_M^{N-L-1} \end{bmatrix}, \quad (2.151)$$

onde  $L$  representa um janelamento nos dados na recepção.

$$\mathbf{Z}_R = \begin{bmatrix} z_1^{L-1} & z_1^{L-2} & \dots & 1 \\ z_2^{L-1} & z_2^{L-2} & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ z_M^{L-1} & z_M^{L-2} & \dots & 1 \end{bmatrix}, \quad (2.152)$$

Com isso, pode-se escrever

$$\mathbf{X} = \mathbf{Z}_L \mathbf{B} \mathbf{Z}_R, \quad (2.153)$$

$$\mathbf{Y} = \mathbf{Z}_L \mathbf{B} \mathbf{Z} \mathbf{Z}_R, \quad (2.154)$$

onde  $\mathbf{B}$  é uma matriz diagonal, de tamanho  $M \times M$  contendo os valores de  $b_i$ , enquanto  $\mathbf{Z}$  também é uma matriz de dimensões  $M \times M$ , diagonal, contendo os valores de  $z_i$ . A partir dessas definições, o parâmetro  $L$  é chamado de *parâmetro do pencil*, enquanto os elementos  $z_t$  estão relacionados com a redução de posto de  $\mathbf{Y}$  em relação a  $\mathbf{X}$ .

Considere o MP  $(\mathbf{Y} - z_t \mathbf{X})$ . Deseja-se descobrir os autovetores nulos desse *matrix pencil*, ou seja, achar os vetores  $\mathbf{q}_t$  tais que  $(\mathbf{Y} - z_t \mathbf{X})\mathbf{q}_t = \mathbf{0}$ , ou, numa abordagem LS,

$$\mathbf{X}^\dagger \mathbf{Y} \mathbf{q}_t = z_t \mathbf{X}^\dagger \mathbf{X} \mathbf{q}_t = z_t \mathbf{q}_t, \quad (2.155)$$

onde  $\dagger$  representa a pseudo-inversa de Moore-Penrose.

Prosseguindo o desenvolvimento, agora considerando AWGN presente, faz-se uma SVD sobre  $\mathbf{X}$ ,

$$\mathbf{X} = \mathbf{U}_0 \mathbf{\Sigma}_0 \mathbf{V}_0^H. \quad (2.156)$$

Conseqüentemente,  $\mathbf{X}^\dagger = \mathbf{V}_0 \mathbf{\Sigma}_0^{-1} \mathbf{U}_0^H$ . Então,

$$\mathbf{X}^\dagger \mathbf{Y} \mathbf{q}_t = \mathbf{V}_0 \mathbf{\Sigma}_0^{-1} \mathbf{U}_0^H \mathbf{Y} \mathbf{q}_t = z_t \mathbf{q}_t. \quad (2.157)$$

Como  $\mathbf{q}_t = \mathbf{V}_0 \mathbf{V}_0^H \mathbf{q}_t$  e  $\mathbf{V}_0^H \mathbf{V}_0 = \mathbf{I}$ ,

$$\mathbf{V}_0 \mathbf{\Sigma}_0^{-1} \mathbf{U}_0^H \mathbf{Y} \mathbf{V}_0 \mathbf{V}_0^H \mathbf{q}_t = z_t \mathbf{V}_0 \mathbf{V}_0^H \mathbf{q}_t, \quad (2.158)$$

Multiplicando-se os dois lados da equação (2.158) por  $\mathbf{V}_0^H$ ,

$$\mathbf{\Sigma}_0^{-1} \mathbf{U}_0^H \mathbf{Y} \mathbf{V}_0 (\mathbf{V}_0^H \mathbf{q}_t) = z_t (\mathbf{V}_0^H \mathbf{q}_t), \quad (2.159)$$

ou seja, os escalares  $z_t$  representam os autovalores de  $\mathbf{\Sigma}_0^{-1} \mathbf{U}_0^H \mathbf{Y} \mathbf{V}_0$ , associados aos autovetores  $\mathbf{V}_0^H \mathbf{q}_t$ . Percebe-se que  $z_t$  então são equivalentes aos autovalores de  $\mathbf{X}^\dagger \mathbf{Y}$ .

O problema que utiliza a modelagem clássica para estimação de DoA, apresentado nas seções anteriores, tem uma relação com o problema de estimação de  $z_t$  na modelagem apresentada nesta seção. A diferença básica nas modelagens é que a estimação de DoA tenta achar valores para modulação de fase para a frequência

$\omega_t$ . Além disso, o fator de desvanecimento  $e^{\alpha t}$  da modelagem apresentada na equação (2.149) não está incluído na modelagem de DoA assumido nos algoritmos propostos nesta tese.

## 2.5 Discussões do Capítulo

**Em relação a todos os algoritmos descritos:** Este capítulo tem como objetivo mostrar os principais algoritmos que influenciaram o desenvolvimento dos algoritmos propostos nesta tese. Os algoritmos apresentados neste capítulo são mostrados na Figura 2.10.

Os dois ambientes apresentados nesta seção, ou seja, equalização cega e estimação de DoA, apresentam alguns pontos em comum. Como visto na Seção 2.2, devido às propriedades de cicloestacionaridade, a modelagem de equalização feita através de TSRs gera um sistema equivalente com múltiplas entradas e saídas, do mesmo modo que o ambiente de múltiplas antenas. Isso proporciona um conjunto de relações entre os ambientes de equalização cega e estimação de DoA que serão detalhadas na Subseção 3.2.

Neste capítulo foram apresentados dois tipos de algoritmos para estimação de DoA: algoritmos com base no método ESPRIT (com variações *beam-space* e *unitary*) e o algoritmo MP. O método ESPRIT foi apresentado por representar um dos tipos de algoritmos mais usados atualmente para estimação paramétrica de DoA. O método MP foi apresentado pela semelhança nas estruturas utilizadas em relação ao algoritmo proposto CB-DoA, a ser apresentado na Subseção 3.3. No entanto, o método MP apresenta restrições mais rígidas na geometria das antenas quando comparado com o algoritmo proposto. Os métodos com base em ESPRIT são os que apresentam as mesmas restrições do algoritmo proposto, e será portanto, o método utilizado para comparação de complexidade computacional e MSE (*mean square error*) no Capítulo 3. As versões *beam-space* e *unitary* são apresentadas por serem utilizadas como comparação para algoritmos propostos, respectivamente, nas Subseções 3.4 e 3.5.

**As Redundâncias presentes no ESPRIT:** A partir da descrição do algoritmo ESPRIT, pode-se verificar a presença de algumas redundâncias em suas estruturas.

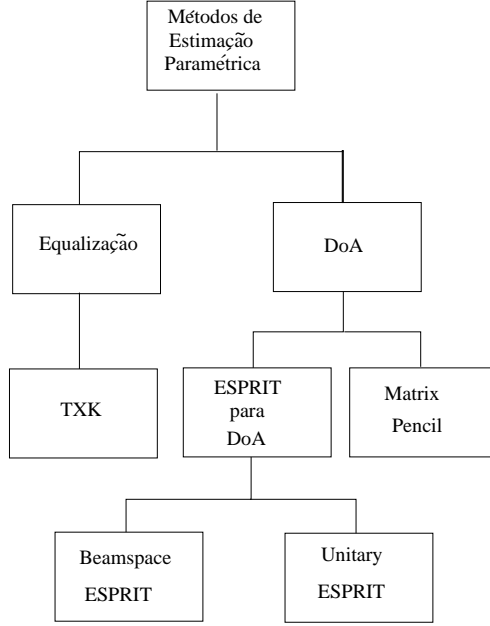


Figura 2.10: Representação dos algoritmos apresentados no Capítulo 2, compreendendo os ambientes de equalização cega e estimação de DoA.

Como visto na equação (2.62), sua primeira decomposição é feita utilizando-se a matriz  $\mathbf{R}_z$ . Pode-se verificar que a estrutura de  $\mathbf{R}_z$  apresenta a seguinte forma

$$\mathbf{R}_z = \begin{bmatrix} \mathbf{R}_x & \mathbf{R}_{xy} \\ \mathbf{R}_{yx} & \mathbf{R}_y \end{bmatrix}. \quad (2.160)$$

Usando-se as matrizes de seleção  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  conforme definidas na Subseção 2.3.4, pode-se dizer que

$$\mathbf{R}_x = \mathbf{J}_X \mathbf{R}_z \mathbf{J}_X^H = \mathbf{A}_X \mathbf{R}_s \mathbf{A}_X^H + \sigma^2 \mathbf{J}_X \mathbf{\Sigma}_n \mathbf{J}_X^H, \quad (2.161)$$

$$\mathbf{R}_{yx} = \mathbf{J}_Y \mathbf{R}_z \mathbf{J}_X^H = \mathbf{A}_X \mathbf{\Phi} \mathbf{R}_s \mathbf{A}_X^H + \sigma^2 \mathbf{J}_Y \mathbf{\Sigma}_n \mathbf{J}_X^H, \quad (2.162)$$

$$\mathbf{R}_{xy} = \mathbf{J}_X \mathbf{R}_z \mathbf{J}_Y^H = \mathbf{A}_X \mathbf{R}_s \mathbf{\Phi}^H \mathbf{A}_X^H + \sigma^2 \mathbf{J}_X \mathbf{\Sigma}_n \mathbf{J}_Y^H, \quad (2.163)$$

$$\mathbf{R}_y = \mathbf{J}_Y \mathbf{R}_z \mathbf{J}_Y^H = \mathbf{A}_X \mathbf{\Phi} \mathbf{R}_s \mathbf{\Phi}^H \mathbf{A}_X^H + \sigma^2 \mathbf{J}_Y \mathbf{\Sigma}_n \mathbf{J}_Y^H. \quad (2.164)$$

Considera-se agora o caso em que os dados são transmitidos por fontes decorrelacionadas, ou seja,  $\mathbf{R}_s$  é diagonal, o que é uma hipótese bastante razoável no caso de transmissão com multiplexação espacial. Mesmo no caso de transmissão com diversidade, pode-se usar métodos de suavização espacial, como o mostrado em [39] para se decorrelacionar os dados das fontes. Considera-se também que

$$\mathbf{J}_X \mathbf{\Sigma}_n \mathbf{J}_X^H = \mathbf{J}_Y \mathbf{\Sigma}_n \mathbf{J}_Y^H, \quad (2.165)$$

ou seja, que o campo de ruído nas antenas do *sub-array* X tenha o mesmo comportamento estatístico de segunda ordem das antenas do *sub-array* Y. Nesse caso,

$$\begin{aligned}
\mathbf{R}_y &= \mathbf{A}_X \mathbf{R}_s \Phi \Phi^H \mathbf{A}_X^H + \sigma^2 \mathbf{J}_Y \Sigma_n \mathbf{J}_Y^H \\
&= \mathbf{A}_X \mathbf{R}_s \mathbf{A}_X^H + \sigma^2 \mathbf{J}_X \Sigma_n \mathbf{J}_X^H \\
&= \mathbf{R}_x,
\end{aligned} \tag{2.166}$$

ou seja, sob tais hipóteses,  $\mathbf{R}_y = \mathbf{R}_x$ . Com isso, verifica-se que o processamento sobre  $\mathbf{R}_z$  poderia ser feito utilizando-se apenas  $\mathbf{R}_x$  e  $\mathbf{R}_{xy}$ . Esse fato será explorado nos algoritmos propostos que são apresentados nos Capítulos 3 e 4.

**Vantagens apresentadas por sistemas com múltiplas antenas:** O uso de um receptor com múltiplas antenas, como os descritos neste capítulo, apresenta uma série de vantagens em relação a receptores móveis tradicionais com uma única antena. Na seção 1.2 é comentado o uso do receptor com múltiplas antenas no combate à CCI, que é considerada uma das principais vantagens desse tipo de receptor. Além do combate à CCI, em [9] são apresentadas as seguintes vantagens:

- Redução no *delay spread* e no combate ao desvanecimento de multipercurso (*multipath fading*) [8]: essas interferências são combatidas pelo uso das informações de diferentes antenas, produzindo uma diversidade espacial de estimativas para desvanecimento;
- Atribuição dinâmica de canais: aproveitando-se o efeito de um conjunto de antenas no receptor sobre a geometria da célula, que pode variar ao longo do tempo, de acordo com as necessidades ligadas ao tráfego de informação;
- Redução na quantidade de *handoffs*: mais uma vez por causa da possibilidade de flexibilização na geometria das células.

Além desses itens, é mostrado que outras vantagens aparecem como consequência das melhorias anteriores, como a redução na probabilidade de falha de transmissão (*outage*), causada pela diminuição da CCI, e uma implementação com maior eficiência e menor custo, por causa da flexibilidade e robustez do sistema com múltiplas antenas.

As vantagens apresentadas acima têm como foco sistemas com multiplexação de acesso feita por TDMA (*time division multiple access*). Como foi mencionado na

Subseção 1.2, o uso de múltiplas antenas de recepção em um sistema de comunicações permite que o compartilhamento de acesso possa ser feito através de multiplexação espacial. Isso dá origem aos sistemas SDMA (*space division multiple access*), que permitem que todos os usuários transmitam simultaneamente utilizando a mesma banda, sem a redundância de informação presente em novos esquemas de compartilhamento de acesso como a transmissão com códigos de espalhamento CDMA [11].

Na tese [40], é desenvolvida uma modelagem para um sistema com *array* de antenas no *uplink* de um sistema CDMA, apresentando um receptor *Rake* do tipo utilizado pelo IS-95 (*interim standard 95*) modificado de modo a utilizar informações espaciais dos sinais transmitidos. Verifica-se que o novo receptor *Rake* apresenta um desempenho superior ao *Rake* tradicional. Em vários cenários de simulação utilizados, o sistema com *array* de antenas apresenta uma menor BER (*bit error rate*).

Uma discussão ampla de vantagens da utilização de *array* de antenas em sistemas comerciais IS-95 e o IS-136 (*interim standard 136*), que é baseado em TDMA, pode ser encontrada na tese [41]. O trabalho tem como foco as vantagens relacionadas a aumento da capacidade do sistema e redução na BER.

**Utilização de estimação de DoA em sistemas padronizados:** O desenvolvimento de técnicas de estimação de DoA e as vantagens vislumbradas pela utilização de *arrays* de antenas [9] levaram à incorporação de múltiplas antenas em alguns sistemas padronizados. Em [42] o algoritmo MUSIC é simulado em um ambiente híbrido TDMA-SDMA, desenvolvido a partir de modificações no padrão DECT (*digital enhanced cordless communications*) de telefonia portátil (*cordless*). Resultados apresentados em [42] mostram que, com a utilização de técnicas de supressão de interferência, como transmissão descontínua, o sistema com múltiplas antenas consegue uma qualidade de serviço superior ao padronizado.

Em [43], é proposto um desenvolvimento do padrão DECT para atender às especificações de telefonia de 3ª geração feito pelo 3GPP (*3<sup>rd</sup> generation partnership project*), comitê que padronizou os requisitos para um padrão continental europeu de telefonia móvel em meados da década de 1990. O padrão DECT é parte da família de padrões IMT-2000 (*international mobile telecommunications 2000*). A evolução do padrão DECT foi desenvolvida pelo consórcio TSUNAMI (*technology*

*in smart antenna for the universal advanced mobile infrastructure*), utilizando um híbrido TDMA-SDMA para multiplexação de acesso. Para se fazer o rastreamento (*tracking*) da posição das fontes é feita uma modificação do MUSIC, com o uso de um filtro de Kalman para se estimar a evolução temporal da estimação de DoA. Ainda no artigo [43], também é implementado um sistema híbrido CDMA-SDMA que atende às especificações de 3ª geração, com o algoritmo MUSIC para estimação de DoA.

Na tese [40] são implementados dois sistemas de estimação de DoA num cenário IS-95 com *array* de antenas. Os métodos implementados são o algoritmo MP e um algoritmo que usa sequências de treinamento nas portadoras-piloto, ambos para realizar a estimação de DoA. Surpreendentemente o método MP, mesmo sem um sinal de referência, apresenta melhor desempenho.

No artigo [44], o algoritmo ESPRIT é implementado no *uplink* de um sistema padronizado WCDMA (*wideband code division multiple access*), de acordo com as especificações do 3GPP. A estimação de DoA é feita para diversos tipos de canal em taxas de até 4096 kcps (*kilochips per second*). Em outro artigo dos mesmos autores [45], é feita uma comparação de desempenho entre vários algoritmos de estimação de DoA no *uplink* do WCDMA para um *array* receptor com 8 antenas. Para um cenário em que o número de usuários transmissores já é conhecido, o algoritmo com melhor desempenho nas simulações em termos de taxa de erro foi o *Unitary* ESPRIT [45].

O artigo [46] aplica os algoritmos MUSIC e ESPRIT, além de um algoritmo proposto para estimação de DoA em conjunto com estimação de canal no *uplink* de um sistema CDMA contendo um receptor *rake* e um detector multiusuário MMSE.

## 2.6 Conclusões do Capítulo

Neste capítulo foram apresentados os dois principais métodos cujas ideias originaram os novos algoritmos propostos para estimação. O método TXK é o primeiro algoritmo de equalização cega de canais baseado apenas em estatísticas de segunda ordem do sinal recebido. Por sua vez, o método ESPRIT é um método tradicional de estimação paramétrica, originalmente proposto num cenário de estimação de DoA



para um ambiente com múltiplas antenas de recepção. São também apresentadas as versões *beamspace* e *unitary* para o método ESPRIT, que diminuem sua complexidade computacional. Outro algoritmo de estimação de DoA apresentado é o MP, cuja solução está relacionada com a dos algoritmos a serem usados no próximo capítulo.

Na parte de discussões do capítulo, é mostrado como o algoritmo ESPRIT trabalha com dados que apresentam redundância. A abordagem com base em TXK a ser desenvolvida no próximo capítulo explora essa redundância para diminuir a complexidade computacional do ESPRIT. Além disso, foram discutidas algumas implementações presentes na literatura de algoritmos de estimação de DoA em sistemas padronizados. Também são discutidas as vantagens do uso de *array* de antenas em sistemas de comunicação.

A partir dos algoritmos e aplicações mostrados neste capítulo foi feito o desenvolvimento dos novos algoritmos propostos para estimação de DoA, mostrados no próximo capítulo.

# Capítulo 3

## Novos Algoritmos para Estimação de DoA

### 3.1 Introdução do Capítulo

Neste capítulo serão apresentados novos algoritmos para estimação baseados nos algoritmos TXK e ESPRIT aplicados ao contexto de estimação de DoA em sistemas com múltiplas antenas de recepção [2]. O objetivo para o desenvolvimento desses algoritmos é ter uma complexidade computacional menor que a do algoritmo ESPRIT utilizando-se as mesmas restrições geométricas.

São apresentados quatro algoritmos de estimação de DoA ao longo deste capítulo:

- *CB-DoA (covariance-based direction-of-arrival estimation)*: primeiro algoritmo desenvolvido, uma alternativa de baixa complexidade para a versão de covariância do ESPRIT para estimação de DoA;
- *BeamSpace CB-DoA*: diminui a complexidade do CB-DoA através da utilização de projeções em subespaços de menor dimensão, os *beamSpace* DFTs;
- *Real CB-DoA*: diminui a complexidade do CB-DoA ao gerar um algoritmo com predomínio de operações reais. Tal vantagem é obtida ao custo de se ter uma maior restrição na geometria do *array* de recepção;
- *DD-DoA (direct data direction-of-arrival estimation)*: alternativa de baixa complexidade para a versão SVD (*single value decomposition*) do ESPRIT.

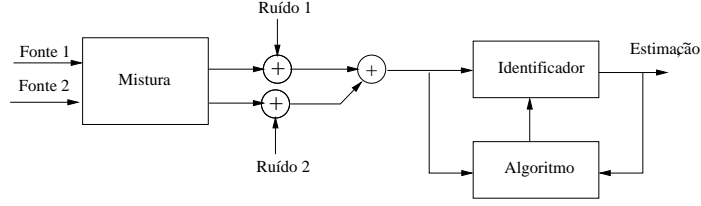


Figura 3.1: Ambiente de identificação que engloba, como casos particulares, os ambientes de equalização cega e de estimação de DoA.

## 3.2 Estendendo o TXK para a estimação de DoA

Nesta seção, o ambiente de equalização é estendido de modo a englobar uma configuração mais geral relacionada tanto à equalização quanto à estimação de DoA. A partir disso, o algoritmo TXK será estendido de modo a possibilitar que possa ser feita tanto a equalização quanto a estimação de DoA. Esta subseção tem como objetivo mostrar o raciocínio implícito na tarefa de desenvolver um algoritmo com base no TXK para estimação de DoA.

Como apresentado na Subseção 2.2.2, uma entrada  $s(k)$  possuindo propriedades de cicloestacionaridade é equivalente a um sistema multicanal estacionário com entrada  $\mathbf{s}(k)$ , de comprimento  $Q$ , mostrada na equação (2.3), reproduzida a seguir

$$\mathbf{s}(k) = \begin{bmatrix} s(kQ) \\ s(kQ + 1) \\ \vdots \\ s(kQ + Q - 1) \end{bmatrix}, \quad (3.1)$$

onde  $Q$  representa o comprimento das TSRs, e corresponde também ao número de subcanais representados.

Nesse sistema equivalente com múltiplos canais, a covariância cruzada entre as entradas  $s(kQ + i), 0 \leq i \leq (Q - 1)$  dos diversos subcanais é predefinida pela covariância temporal da sequência cicloestacionária  $s(k)$ . No caso do algoritmo TXK, o sinal  $s(k)$  é modelado como branco, o que corresponde a um  $\mathbf{s}(k)$  com covariância  $E[\mathbf{s}(k)\mathbf{s}^H(k)]$  nula para elementos fora de diagonal principal, isto é, representando a covariância entre instantes distintos. Generaliza-se agora o ambiente original do algoritmo TXK a fim de que possam existir sinais com covariância cruzada entre subcanais mais genérica do que a definida pela covariância temporal de cada si-

nal. No novo ambiente, sinais de entrada de cada subcanal não são necessariamente amostras temporais de um mesmo processo estocástico, ou seja, gera-se  $s_i(k)$  em cada  $i$ -ésimo subcanal, sendo os sinais de diferentes subcanais relacionados por uma matriz de covariância com estrutura conhecida.

Considera-se inicialmente que existam dois sinais de entrada:  $s_1(k)$ , com autocovariância  $\mathbf{R}_{s_1} = E[\mathbf{s}_1(k)\mathbf{s}_1^H(k)]$ ; e  $s_2(k)$  com autocovariância  $\mathbf{R}_{s_2}$ . Esses sinais estão relacionados por uma matriz de covariância cruzada  $\mathbf{R}_{s_{12}} = E[\mathbf{s}_1(k)\mathbf{s}_2^H(k)]$ . O sistema de mistura gera duas saídas, corrompidas respectivamente pelos ruídos  $n_1(k)$  e  $n_2(k)$ . Tal sistema é resumidamente descrito na Figura 3.1, que representa uma generalização do sistema da Figura 2.1. O ambiente da Figura 3.1 é então modelado como dois sistemas com entradas  $\mathbf{s}_1$  e  $\mathbf{s}_2$ . A entrada  $\mathbf{s}_1$  passará por um sistema representado pela matriz de mistura  $\mathbf{M}_1$  e sofrerá a interferência do ruído  $\mathbf{n}_1$ , gerando o sinal  $\mathbf{x}_1$ . De maneira equivalente, a entrada  $\mathbf{s}_2$  do sistema  $\mathbf{M}_2$  gera uma saída  $\mathbf{x}_2$  após sofrer a interferência de  $\mathbf{n}_2$ . Matematicamente, isso pode ser modelado pelas seguintes equações

$$\mathbf{x}_1 = \mathbf{M}_1\mathbf{s}_1 + \mathbf{n}_1, \quad (3.2)$$

$$\mathbf{x}_2 = \mathbf{M}_2\mathbf{s}_2 + \mathbf{n}_2. \quad (3.3)$$

Os sistemas  $\mathbf{M}_1$  e  $\mathbf{M}_2$  estão relacionados linearmente por

$$\mathbf{M}_2 = \mathbf{\Phi}\mathbf{M}_1, \quad (3.4)$$

onde  $\mathbf{\Phi}$  é uma matriz que representa uma redundância que existe na representação do sistema, sendo também o conjunto de parâmetros do sistema a ser estimado. De maneira alternativa, o sistema pode ser representado por

$$\mathbf{z} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \mathbf{M} \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \end{bmatrix}. \quad (3.5)$$

Como visto na Subseção 2.2.2, no algoritmo TXK original, há apenas uma matriz de mistura  $\mathbf{H}$ , que representa as TSRs associadas ao canal original, enquanto o conjunto de parâmetros estimado no TXK está relacionado ao autovetor associado ao maior autovalor da matriz de autocovariância sem ruído de *lag* 1  $\mathbf{R}_x(1) = \mathbf{H}\mathbf{L}\mathbf{H}^H$ . O processo de estimação paramétrica se conclui na equação (2.24) do TXK. As etapas seguintes do TXK envolvem a estimação  $\hat{\mathbf{s}}(k)$  do sinal transmitido, e são

manipulações sobre o parâmetro estimado, de maneira equivalente à etapa SSI4. Do mesmo modo, uma versão do TXK para estimação de DoA exigiria a 4ª etapa de estimação SSI, isto é, identificação das DoAs a partir de  $\hat{\Phi}$ , com formulação diferente do que é feito para o problema de equalização.

**Estimação neste novo contexto:** Consideram-se estimadas a autocovariância  $\mathbf{R}_{x1}$  de  $\mathbf{x}_1$  e  $\mathbf{R}_{n1}$  como a autocovariância do ruído  $\mathbf{n}_1$ . São realizadas então as primeiras duas etapas de estimação, de acordo com as definições da Subseção 2.3.1,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}, \quad (3.6)$$

$$[\mathbf{U}_s, \Lambda_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_{x1}, \mathbf{R}_{n1}, \text{SSI1}(\mathbf{Z})), \quad (3.7)$$

Consideram-se as seguintes matrizes de covariâncias cruzadas,

$$\mathbf{R}_b = E[\mathbf{x}_1 \mathbf{x}_2^H], \quad (3.8)$$

$$\Sigma_{12} = E[\mathbf{n}_1 \mathbf{n}_2^H]. \quad (3.9)$$

Aplica-se então uma operação TB sobre  $(\mathbf{R}_b - \Sigma_{12})$ , ou seja,

$$[\Psi] = \text{TB}(\mathbf{U}_s, \Sigma_s, \mathbf{R}_b - \Sigma_{12}). \quad (3.10)$$

de modo que

$$\Psi = \Lambda_s^{-1} \mathbf{U}_s^H (\mathbf{R}_b - \Sigma_{12}) \mathbf{U}_s \Lambda_s = \Lambda_s^{-1} \mathbf{U}_s^H \mathbf{M}_1 \mathbf{R}_{s12} \Phi \mathbf{M}_1^H \mathbf{U}_s \Lambda_s, \quad (3.11)$$

Então, a matriz diagonal  $\Phi$  pode ser obtida a partir de uma EVD sobre a matriz  $\Psi$  como mostrado na equação (3.11).

Por fim, é efetuada a quarta etapa de estimação SSI para DoA, de acordo com a equação (2.54), ou seja,

$$[\hat{\theta}] = \text{SSI4}(\hat{\Phi}). \quad (3.12)$$

A formulação do TXK estendido é o embrião do desenvolvimento do algoritmo teórico GTXK (*generalized TXK*) que pretende criar um *framework* unificado para os algoritmos paramétricos de equalização cega, estimação de DoA e estimação de CFO. O *framework* unificado será mostrado na Seção 4.6.

### 3.3 Covariance-Based DoA Estimation (CB-DoA)

Considere o cenário de estimação de DoA, conforme é apresentado na Subseção 2.3.3, ou seja, representado pelas equações (2.52) e (2.53), repetidas aqui por conveniência,

$$\mathbf{x}(k) = \mathbf{A}_X \mathbf{s}(k) + \mathbf{n}_x(k), \quad (3.13)$$

$$\mathbf{y}(k) = \mathbf{A}_X \Phi \mathbf{s}(k) + \mathbf{n}_y(k). \quad (3.14)$$

Considera-se o ambiente de equalização estendido de modo a englobar o problema de DoA. Tal ambiente é definido na Subseção 3.2. Podem ser verificadas as equivalências mostradas na Tabela 3.1.

Tabela 3.1: Equivalências entre TXK estendido e o ambiente de estimação de DoA.

DoA	TXK Estendido
$\mathbf{x}(k)$	$\mathbf{x}_1$
$\mathbf{y}(k)$	$\mathbf{x}_2$
$\mathbf{n}_x(k)$	$\mathbf{n}_x$
$\mathbf{n}_y(k)$	$\mathbf{n}_y$
$\mathbf{A}_X$	$\mathbf{M}_1$

Definem-se então algumas estruturas de covariância dos sinais recebidos

$$\mathbf{R}_x = E[\mathbf{x}(k)\mathbf{x}^H(k)], \quad (3.15)$$

$$\mathbf{R}_{xy} = E[\mathbf{x}(k)\mathbf{y}^H(k)], \quad (3.16)$$

$$\sigma_N^2 \mathbf{R}_{nx} = E[\mathbf{n}_x(k)\mathbf{n}_x^H(k)], \quad (3.17)$$

$$\mathbf{R}_{nxy} = E[\mathbf{n}_x(k)\mathbf{n}_y^H(k)]. \quad (3.18)$$

São feitas as etapas 1 e 2 de uma estimação SSI, de acordo com o que foi definido na Subseção 2.3.1, ou seja,

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_x, \mathbf{R}_{n,x}, \text{SSI1}(\mathbf{Z})). \quad (3.19)$$

A equação (3.19) utiliza o par  $\{\mathbf{R}_x, \mathbf{R}_{n,x}\}$  para fazer a segunda etapa de estimação SSI, o que é bem menos custoso computacionalmente que o par  $\{\mathbf{R}_z, \mathbf{R}_n\}$  do ESPRIT, como mostrado na equação (2.64).

Considera-se  $\mathbf{U}_s$  como a matriz contendo os autovetores generalizados correspondentes ao espaço dos sinais em suas colunas. A matriz  $\mathbf{R}_s$  contém os autovalores associados a esses autovetores. Então, existe uma matriz de posto completo  $\mathbf{V}$  tal que

$$\mathbf{A}_X = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{V}. \quad (3.20)$$

Sabendo-se que

$$\mathbf{R}_{xy} = \mathbf{A}_X \mathbf{\Phi}^H \mathbf{A}_X^H + \mathbf{R}_{nxy}, \quad (3.21)$$

é realizada uma operação TB, conforme definida na Subseção 2.2.2,

$$[\mathbf{\Psi}] = \text{TB}(\mathbf{U}_s, \mathbf{\Lambda}_s, \mathbf{R}_{xy} - \hat{\sigma}^2 \mathbf{R}_{nxy}), \quad (3.22)$$

onde  $\hat{\sigma}^2$  é o valor da potência do ruído estimado a partir dos autovalores de  $\mathbf{R}_x$ . Então, a partir da equação (3.20) e da operação TB tem-se

$$\mathbf{\Psi} = \mathbf{V} \mathbf{\Phi}^H \mathbf{V}^H, \quad (3.23)$$

ou seja, podemos estimar os ângulos de chegada através de uma EVD sobre a matriz  $\mathbf{\Psi}$ . Por fim, é aplicada a função SSI4 para o ambiente de DoA, como definido na equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}), \quad (3.24)$$

de acordo com a equação (2.54), repetida a seguir, por conveniência,

$$\hat{\theta}_i = \arcsen\left(\frac{c}{j\omega\Delta} \ln(\hat{\phi}_i)\right). \quad (3.25)$$

O algoritmo descrito nesta seção é denominado *Covariance-Based Direction of Arrival Estimation* (CB-DoA). A Figura 3.2 contém um resumo das operações do algoritmo CB-DoA.

### 3.3.1 Comparações entre CB-DoA e TLS-ESPRIT

**Complexidade Computacional:** Mais uma vez utilizou-se uma notação inspirada em códigos Matlab, a Tabela 3.2 compara os algoritmos TLS-ESPRIT e CB-DoA apresentados, respectivamente, na Subseção 2.3.3 e na Seção 3.3. São definidas as seguintes matrizes de seleção

$$\mathbf{J}_X = \begin{bmatrix} \mathbf{I}_M & \mathbf{0}_M \end{bmatrix} \quad (3.26)$$

$$\mathbf{J}_Y = \begin{bmatrix} \mathbf{0}_M & \mathbf{I}_M \end{bmatrix} \quad (3.27)$$

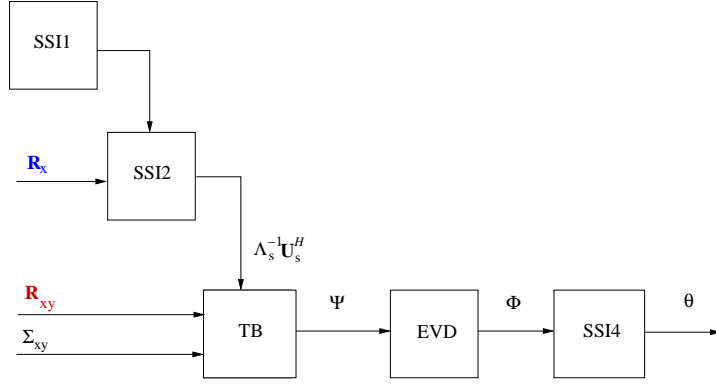


Figura 3.2: Diagrama de blocos representando o processo de estimação do algoritmo CB-DoA.

A partir dessa tabela, verifica-se que o algoritmo TLS-ESPRIT utiliza:

- 1 GEVD de um par de matrizes de dimensões  $2N \times 2N$ ;
- 2 EVDs (1 para uma matriz hermitiana de dimensões  $2M \times 2M$  e 1 para uma matriz não-hermitiana de dimensões  $M \times M$ );
- 1 inversão de uma matriz de dimensões  $M \times M$ ;
- 6 multiplicações de matrizes (5 para pares de matrizes de dimensões  $M \times M$  e 1 para o produto de uma matriz  $2N \times 2N$  por uma matriz  $2N \times M$ ).

Por sua vez, o algoritmo CB-DoA utiliza:

- 1 GEVD de um par de matrizes de dimensões  $N \times N$ ;
- 1 EVD de uma matriz de dimensões  $N \times N$ ;
- 1 inversão de uma matriz diagonal de dimensões  $M \times M$ ;
- 3 multiplicações de matrizes de dimensões  $M \times M$  por matrizes de dimensões  $M \times N$ ;
- 1 subtração de matrizes de dimensões  $N \times N$ .

Apesar de o custo computacional ser bastante dependente da implementação utilizada, verifica-se que o algoritmo CB-DoA requer menos operações que o TLS-ESPRIT. Isso pode ser também verificado na Tabela 3.3, a partir de complexidades



Tabela 3.2: Comparação entre os algoritmos TLS-ESPRIT e CB-DoA.

ESPRIT	CB-DoA
$[\mathbf{U}_s, \mathbf{\Lambda}_s, \hat{\sigma}^2] = \text{GEVD}(\hat{\mathbf{R}}_z(0), \hat{\Sigma}_n(0))$	$[\mathbf{U}_s, \mathbf{\Lambda}_s, \hat{\sigma}^2] = \text{GEVD}(\hat{\mathbf{R}}_x(0), \hat{\mathbf{R}}_{n,x}(0))$
$\mathbf{E}_s = \mathbf{\Sigma}_n(0)\mathbf{U}_s$	$\mathbf{F} = \mathbf{\Sigma}_s^{-1}\mathbf{U}_s^H$
$\mathbf{E}_x = \mathbf{J}_X\mathbf{E}_s$	$\mathbf{R}_a = \hat{\mathbf{R}}_{xy}(0) - \hat{\sigma}_{xy}^2\hat{\mathbf{R}}_{n,xy}(0)$
$\mathbf{E}_y = \mathbf{J}_Y\mathbf{E}_s$	
$\mathbf{E}_a = \begin{bmatrix} \mathbf{E}_x^H \\ \mathbf{E}_y^H \end{bmatrix} \begin{bmatrix} \mathbf{E}_x & \mathbf{E}_y \end{bmatrix}$	
$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{E}_a)$	$\mathbf{\Psi} = \mathbf{F}\mathbf{R}_a\mathbf{F}^H$
$\mathbf{E}_{12} = \mathbf{J}_X\mathbf{E}\mathbf{J}_Y^T$	$[\mathbf{V}, \mathbf{\Phi}^H] = \text{EVD}(\mathbf{\Psi})$
$\mathbf{E}_{22} = \mathbf{J}_Y\mathbf{E}\mathbf{J}_Y^T$	
$\mathbf{\Psi} = -\mathbf{E}_{12}\mathbf{E}_{22}^{-1}$	
$[\mathbf{T}, \mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$	

assintóticas para operações matriciais apresentadas em [27]. Percebe-se que, apesar de exigir o conhecimento, ou estimação das matrizes de covariância de ruído, o algoritmo CB-DoA tem requisitos computacionais bem menos rigorosos que o método tradicional TLS-ESPRIT.

**Desempenho** A fim de se medir o impacto da eficiência computacional, foram realizadas simulações em Matlab para o TLS-ESPRIT e o CB-DoA. Os parâmetros utilizados foram:

- Dados transmitidos gerados a partir de uma distribuição gaussiana com variância 0,5;
- Matriz  $\mathbf{A}_X$  gerada aleatoriamente, cada entrada com distribuição gaussiana de variância 0,5;
- Métrica utilizada: variação de SNR (*signal-to-noise ratio*) em relação ao MSE:

$$\text{MSE} = \frac{1}{M} \sum_{i=0}^{M-1} |\theta_i - \hat{\theta}_i|^2. \quad (3.28)$$

Tabela 3.3: Comparação para as operações requeridas por TLS-ESPRIT e CB-DoA.

Operação	Compl. [27]	TLS-ESPRIT	CB-DoA
EVD Matrizes Não-Hermitianas	$\mathcal{O}(25n^3)$	2	1
EVD Matrizes Hermitianas	$\mathcal{O}(n^2)$	1	1
Inversão Completa	$\mathcal{O}(2n^3/3)$	2	1
Inversão Matrizes Diagonais	$\mathcal{O}(n)$	–	1
Multipl. de Matrizes	$\mathcal{O}(n^3)$	6	3
Subtração	$\mathcal{O}(n^2)$	–	1

- Para cada fonte, 5000 dados transmitidos;
- Média de MSE sobre 300 realizações no *ensemble*.

Os resultados de simulação para  $M = 4$  antenas de transmissão e  $N = 9$  *doublets* de recepção estão representados na Figura 3.3. Já os resultados para  $M = 7$  antenas de transmissão e  $N = 12$  *doublets* de recepção estão mostrados na Figura 3.4.

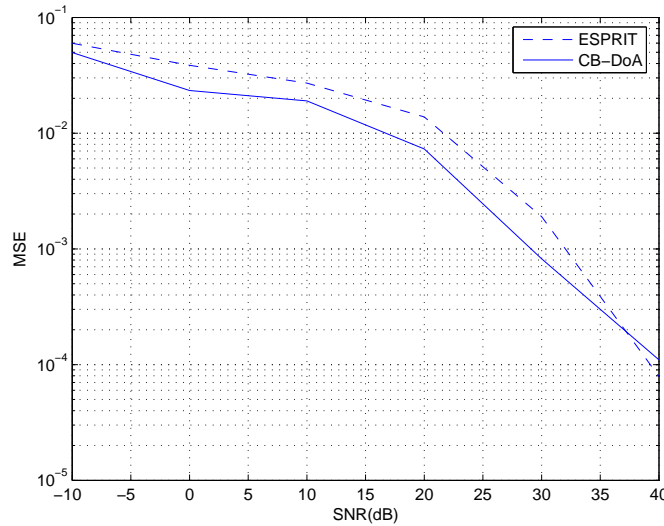


Figura 3.3: MSE para TLS-ESPRIT e CB-DoA em função da SNR para  $M = 4$  antenas de transmissão e  $N = 9$  *doublets* de recepção.

Verifica-se que os MSE são equivalentes em boa parte da faixa de SNRs utilizadas, com o algoritmo CB-DoA apresentando valores de MSE um pouco menores em geral.

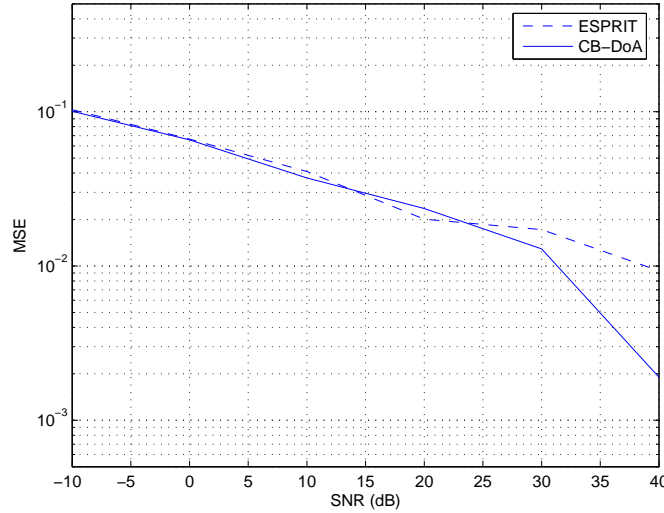


Figura 3.4: MSE para TLS-ESPRIT e CB-DoA em função da SNR para  $M = 7$  antenas de transmissão e  $N = 12$  *doublets* de recepção.

A fim de se verificar o ganho computacional proporcionado na nossa implementação, os algoritmos implementados em Matlab têm seus tempos de estimação medidos na seguinte plataforma:

- PCs com processadores Intel Pentium IV 3GHz;
- Memória RAM 512MB;
- Fedora Linux com Matlab 7.0.

Mais uma vez foram utilizadas 5000 amostras e 300 realizações no ensemble. Os resultados são apresentados na Tabela 3.4. A métrica **Razão** está definida como

$$\mathbf{Razão} = \frac{\text{Tempo para CB-DoA}}{\text{Tempo para ESPRIT}}. \quad (3.29)$$

Observa-se que o algoritmo CB-DoA torna-se cada vez mais rápido que o TLS-ESPRIT à medida em que se aumenta o número de antenas de transmissão e recepção utilizadas, sem qualquer prejuízo em termos de MSE. Destaca-se que, para um número grande antenas, conseguiu-se um tempo de simulação quase 80% menor. Esses resultados de simulação por tempo são apenas ilustrativos, pois possuem uma alta dependência da maneira como são implementados os métodos. Os valores quantitativos sofrem também dependência de ocupação do sistema operacional,

Tabela 3.4: Tempo médio de execução para TLS-ESPRIT and CB-DoA para número distintos de antenas transmissoras ( $M$ ) e *doublets* de recepção ( $N$ ).

$M$	$N$	TLS-ESPRIT	CB-DoA	Razão
1	4	$1,17 \cdot 10^{-3}\text{s}$	$9,36 \cdot 10^{-4}\text{s}$	0,795
4	9	$1,92 \cdot 10^{-3}\text{s}$	$1,28 \cdot 10^{-3}\text{s}$	0,668
7	12	$3,64 \cdot 10^{-3}\text{s}$	$1,67 \cdot 10^{-3}\text{s}$	0,459
14	30	$1,34 \cdot 10^{-2}\text{s}$	$3,28 \cdot 10^{-3}\text{s}$	0,243
18	39	$2,91 \cdot 10^{-2}\text{s}$	$5,80 \cdot 10^{-3}\text{s}$	0,199

portanto, devem ser considerados de maneira diferente que o restante da análise de desempenho e complexidade.

### 3.3.2 CB-DoA e *Cramer-Rao Lower Bound*

No estudo de estimação, um limite teórico é estabelecido para a variância de estimadores, tal limite é conhecido como *Crámer-Rao lower bound* (CRLB) . Tal limite pode ser comparado com o MSE do sistema, que se estabelece como uma estimativa para a variância dos mesmos, quando se supõe média nula para erros de estimações de DoA.

A fim de se realizar um estudo um pouco mais teórico sobre o algoritmo CB-DoA, foi utilizada a expressão simplificada para o CRLB apresentado em [24] para uma única antena de transmissão,

$$\text{CRLB} = \frac{6}{\text{SNR} \cdot N^3 K}, \quad (3.30)$$

onde  $K$  é o número de *snapshots*. Mais uma vez foram utilizados 5000 dados de transmissão e 300 realizações no *ensemble*. Foram utilizadas também  $N = 4$  *doublets* de recepção. Os resultados para os algoritmos TLS-ESPRIT e CB-DoA estão representados na Figura 3.5. Nota-se que ambos algoritmos estão distantes do CRLB aproximado. Isso pode ser atribuído tanto à aproximação do CRLB, quanto ao fato de os algoritmos de subespaço se utilizarem de aproximações em relação aos algoritmos mais complexos computacionalmente baseados em ML. Um terceiro fator que pode ser apontado é a utilização de *array manifolds* com elementos aleatórios,

o que simula alguns DoAs que se encontram em faixas de valores possivelmente problemáticas em termos de resolução para cada um dos algoritmos sem violar as condições para utilização da expressão (3.30).

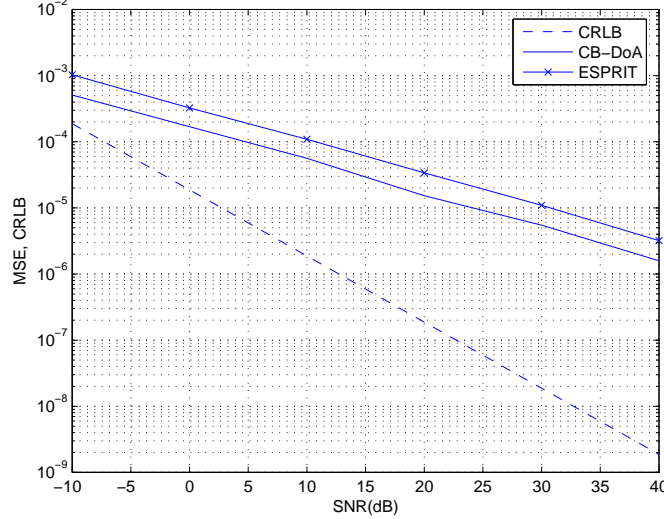


Figura 3.5: MSE para CB-DoA e TLS-ESPRIT, e o CRLB aproximado, com  $M = 1$  e  $N = 4$ .

### 3.4 *Beamspace* CB-DoA

Nesta seção é desenvolvida uma extensão do algoritmo CB-DoA utilizando-se o conceito de *beamspace* DFT e a modelagem de dados apresentados na Subseção 2.3.4. Neste caso, os sinais recebidos são projetados por matrizes  $\mathbf{T}_i$  em espaços diferentes da aquisição, denominados *beamspaces*. Após o processamento, é aplicado o algoritmo CB-DoA para cada subconjunto de dados, como representado na Figura 3.6. O *array* de recepção é um ULA com seus *sub-arrays* são escolhidos com sobreposição, como explicado na Subseção 2.3.4 [3].

Assim como para o *Beamspace* ESPRIT, devem ser mantidas as propriedades de invariância à rotação nos dados a serem processados no *Beamspace* CB-DoA, ou seja, as restrições geométricas de invariância à translação devem ser garantidas. Para isso, necessita-se que exista uma matriz  $\mathbf{Q}$ , que obedeça a equação (2.85), repetida

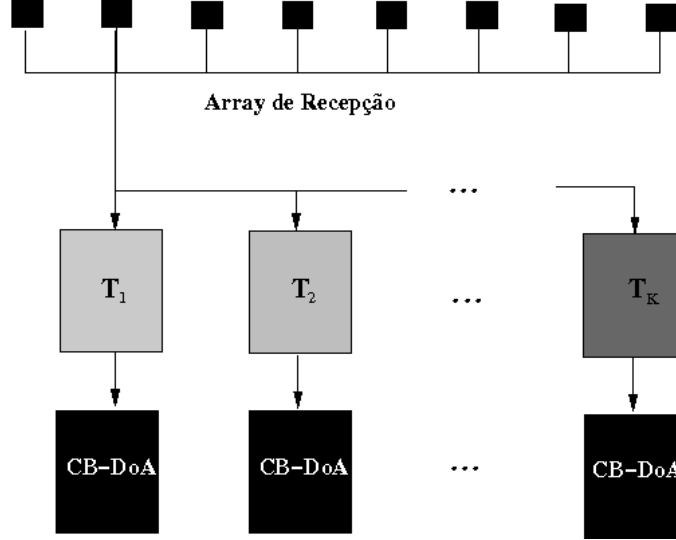


Figura 3.6: Esquema de funcionamento para o algoritmo *BeamSpace* CB-DoA.

a seguir,

$$\begin{cases} \mathbf{Q}\mathbf{F}^H\mathbf{t}_i = 0, & 0 \leq i < N - K, \\ \mathbf{Q}\mathbf{t}_i = 0, & K \leq i < N, \end{cases} \quad (3.31)$$

onde os vetores  $\mathbf{t}_i$  representam as colunas de  $\mathbf{T}_i^H$ .

São definidas mais uma vez, como na equação (2.81), as matrizes  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  que selecionam  $K$  dentre  $N$  colunas de uma matriz,

$$\mathbf{J}_X = \begin{bmatrix} \mathbf{I}_K & \mathbf{0}_{\{K \times (N-K)\}} \end{bmatrix}, \quad (3.32)$$

$$\mathbf{J}_Y = \begin{bmatrix} \mathbf{0}_{\{K \times (N-K)\}} & \mathbf{I}_K \end{bmatrix}, \quad (3.33)$$

A partir da discussão na Subseção 2.3.4, reescreve-se a seguir a equação (2.86)

$$\begin{aligned} \mathbf{Q}\mathbf{T}_i^H\mathbf{A} &= \mathbf{Q}\mathbf{T}_i^H\mathbf{J}_X^T\mathbf{J}_X\mathbf{A} \\ &= \mathbf{Q}\mathbf{T}_i^H\mathbf{J}_X^T\mathbf{J}_Y\mathbf{A}\Phi^H \\ &= \mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{J}_Y^T\mathbf{J}_Y\mathbf{A}\Phi^H \\ &= \mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{A}\Phi^H. \end{aligned} \quad (3.34)$$

Após uma EVD sobre  $\mathbf{R}_x$  e a separação dos subespaços dos sinais e ao ruído, gera-se uma matriz  $\mathbf{U}_s$  cujas colunas contêm o subespaço dos sinais de  $\mathbf{R}_x$ . Tem-se então a equação de invariância no *beamSpace*, já apresentada na Subseção 2.3.4,

equação (2.89),

$$\mathbf{Q}\mathbf{T}_i^H\mathbf{U}_s = \mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{U}_s\mathbf{\Phi}^H. \quad (3.35)$$

Considere então a matriz de covariância  $\mathbf{R}_1$ , tal que

$$\mathbf{R}_1 = \mathbf{Q}\mathbf{R}_z\mathbf{Q}^H = \mathbf{Q}[\mathbf{T}_i^H\mathbf{A}\mathbf{R}_s\mathbf{A}^H\mathbf{T}_i + \sigma_N^2\mathbf{I}]\mathbf{Q}^H, \quad (3.36)$$

**Estimação** Inicialmente são feitas as duas primeiras etapas de uma estimação SSI de acordo com o que foi definido na Subseção 2.3.1, ou seja,

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_1, \text{SSI1}(\mathbf{Z})), \quad (3.37)$$

Considere que exista uma matriz quadrada  $\mathbf{V}$ , com posto completo, que satisfaça

$$\mathbf{Q}\mathbf{T}_i^H\mathbf{A} = \mathbf{U}_s\mathbf{\Lambda}_s\mathbf{V} \quad (3.38)$$

de acordo com as condições dadas por (3.31). Como

$$\begin{aligned} \mathbf{\Psi} &= \mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H\mathbf{Q}\mathbf{F}^H(\mathbf{R}_z - \sigma_N^2\mathbf{I})\mathbf{Q}^H\mathbf{U}_s\mathbf{\Lambda}_s \\ &= \mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H\mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{A}\mathbf{A}^H\mathbf{T}_i\mathbf{Q}^H\mathbf{U}_s\mathbf{\Lambda}_s \\ &= \mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H\mathbf{Q}\mathbf{T}_i^H\mathbf{A}\mathbf{\Phi}\mathbf{A}^H\mathbf{T}_i\mathbf{Q}^H\mathbf{U}_s\mathbf{\Lambda}_s, \end{aligned} \quad (3.39)$$

então é realizada uma operação TB, conforme definida na Subseção 2.2.2,

$$[\mathbf{\Psi}] = \text{TB}(\mathbf{U}_s, \mathbf{\Lambda}_s, \mathbf{Q}\mathbf{F}^H(\mathbf{R}_z - \sigma_N^2\mathbf{I})\mathbf{Q}^H). \quad (3.40)$$

A partir da equação (3.38), conclui-se que

$$\mathbf{\Psi} = \mathbf{V}\mathbf{\Phi}\mathbf{V}^H, \quad (3.41)$$

ou seja, a matriz  $\mathbf{\Phi}$  pode ser encontrada a partir de uma EVD sobre  $\mathbf{\Psi}$ . Em seguida é realizada a 4ª etapa de estimação SSI em cenário de DoA, dada pela equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (3.42)$$

A Figura 3.7 mostra um resumo das operações do algoritmo *BeamSpace* CB-DoA.

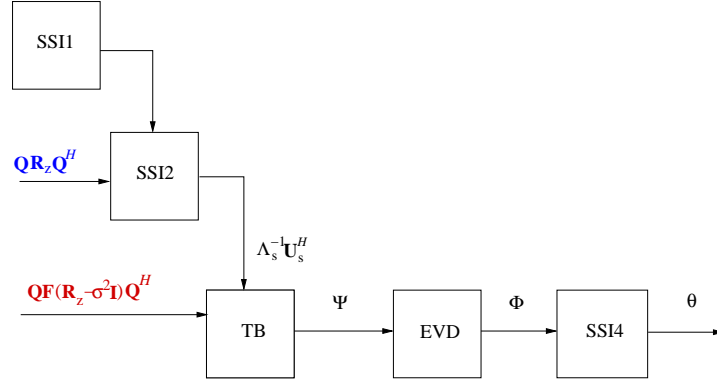


Figura 3.7: Diagrama de blocos representando o processo de estimação do algoritmo *BeamSpace* CB-DoA.

### 3.4.1 Comparações entre *BeamSpace* CB-DoA e *BeamSpace* ESPRIT

**Complexidade Computacional:** A fim de se fazer uma comparação entre o algoritmo *BeamSpace* ESPRIT e a versão *beamSpace* do algoritmo CB-DoA, a complexidade computacional de ambos os algoritmos é mostrada nesta seção. A Tabela 3.5 resume uma comparação das operações básicas de ambos algoritmos. Quando múltiplas linhas ou colunas são necessárias, uma notação inspirada em *scripts* Matlab é utilizada. Lembra-se que  $M$  é o número de fontes,  $N$  o número de sensores e  $L$  é a dimensão do *beamSpace*. Como apresentado em [25], diferentes projeções de *beamSpace* são realizadas em paralelo, devido à inerente modularização do algoritmo. Com isso, a análise de complexidade computacional é realizada para apenas uma sub-banda *beamSpace*. A partir da Tabela 3.5, verifica-se que o algoritmo *BeamSpace* ESPRIT requer:

- 3 EVDs (1 para uma matriz hermitiana de dimensões  $2M \times 2M$ , 1 para uma matriz hermitiana de dimensões  $M \times M$ , e 1 para uma matriz hermitiana de dimensões  $L \times L$ );
- 1 inversão de uma matriz de dimensões  $M \times M$ ;
- 10 multiplicações de matrizes (2 para produtos entre uma matriz com dimensões  $L \times L$  e uma matriz com dimensões  $L \times N$ , 2 para um produto de



Tabela 3.5: Resumo comparativo dos algoritmos *Beamspace* ESPRIT e *Beamspace* CB-DoA.

Beamspace ESPRIT	Beamspace CB-DoA
$[\mathbf{U}_s, \mathbf{\Lambda}_s, \hat{\sigma}_N^2] = \text{EVD}(\hat{\mathbf{R}}_z)$	$[\mathbf{U}_s, \mathbf{\Lambda}_s, \hat{\sigma}_N^2] = \text{EVD}(\mathbf{Q}\hat{\mathbf{R}}_z\mathbf{Q}^H)$
$\mathbf{U}_x = \mathbf{Q}\mathbf{T}_i^H\mathbf{U}_s$	$\mathbf{F} = \mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H$
$\mathbf{U}_y = \mathbf{Q}\mathbf{F}^H\mathbf{T}_i^H\mathbf{U}_s$	$\mathbf{R}_a = \hat{\mathbf{R}}_z - \hat{\sigma}^2\mathbf{I}$
$\mathbf{E}_a = \begin{bmatrix} \mathbf{U}_x^H \\ \mathbf{U}_y^H \end{bmatrix} \begin{bmatrix} \mathbf{U}_x & \mathbf{U}_y \end{bmatrix}$	$\mathbf{\Psi} = \mathbf{F}_s\mathbf{Q}\mathbf{F}^H\mathbf{R}_a\mathbf{Q}^H\mathbf{F}_s^H$
$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{E}_a)$	$[\mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$
$\mathbf{E}_{12} = \mathbf{E}(0 : M - 1, M : \text{end})$	
$\mathbf{E}_{22} = \mathbf{E}(M : \text{end}, M : \text{end})$	
$\mathbf{\Psi} = -\mathbf{E}_{12}\mathbf{E}_{22}^{-1}$	
$[\mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$	

uma matriz de dimensões  $L \times L$  e uma matriz de dimensões  $L \times N$ , 1 para um produto de um par de matrizes de dimensões  $L \times L$ , e 4 para produtos entre uma matriz de dimensões  $M \times L$  e uma matriz com dimensões  $L \times M$ ).

Já o algoritmo *Beamspace* CB-DoA requer:

- 2 EVDs (1 para uma matriz hermitiana de dimensões  $L \times L$  e 1 para uma matriz hermitiana de dimensões  $M \times M$ );
- 1 inversão de uma matriz diagonal de dimensões  $M \times M$ ;
- 8 multiplicações de matrizes (2 para um par de matrizes com dimensões  $L \times L$ , 1 para um produto de uma matriz de dimensões  $M \times M$  e uma matriz de dimensões  $M \times L$ , 4 para produtos entre uma matriz com dimensões  $M \times L$  e uma matriz com dimensões  $L \times M$ , e 1 para produtos de uma matriz de dimensões  $M \times L$  e uma matriz de dimensões  $L \times M$ ).
- 1 subtração entre matrizes de dimensões  $N \times N$  matrices.

Verifica-se que o algoritmo *Beamspace* CB-DoA apresenta uma menor complexidade computacional em relação ao *Beamspace* ESPRIT. De fato, o *Beamspace* CB-DoA exige um número menor de multiplicações de matrizes, e uma operação mais simples de inversão de matriz. Além disso, o *Beamspace* CB-DoA requer menos EVDs, uma operação bastante complexa computacionalmente. A Tabela 3.6 mostra uma comparação mais detalhada em relação à complexidade assintótica das operações básicas, de acordo com [27].

Tabela 3.6: Comparação para as operações requeridas por *beamspace* ESPRIT e *beamspace* CB-DoA.

Operação	Complex. [27]	<i>Beamspace</i> ESPRIT	<i>Beamspace</i> CB-DoA
EVD Matr. Hermitianas	$\mathcal{O}(n^2)$	3	2
Inversão Completa	$\mathcal{O}(2n^3/3)$	1	–
Inversão Matr. Diagonais	$\mathcal{O}(n)$	–	1
Multipl. de Matrizes	$\mathcal{O}(n^3)$	10	8
Subtração	$\mathcal{O}(n^2)$	–	1

**Desempenho:** Assim como na Subseção 3.3.1, foi usado o MSE como métrica de comparação, como definido na equação (3.28). No nosso cenário de comparação, foi computada uma média sobre 200 execuções de Monte Carlo. A Figura 3.8 compara *Beamspace* ESPRIT com o algoritmo proposto *Beamspace* CB-DoA para 2 fontes e 18 antenas de recepção, em função do número de *beams*. Como pode ser observado a partir da Figura 3.8, o comportamento de ambos algoritmos em termos de MSE é bastante próximo, tanto no cenário sem ruído quanto para SNR = 25 dB. Como são usadas 18 antenas de recepção, lembra-se que, para 18 *beams*, os algoritmos *beamspace* equivalem à sua versão *elementspace*. A partir da definição do algoritmo CB-DoA no *beamspace*, percebe-se que o desempenho do algoritmo *Beamspace* CB-DoA em termos de MSE é inferiormente limitado pelo desempenho da sua versão *elementspace*, que equivale ao algoritmo versão *beamspace* quando 18 antenas não usadas, tanto na versão sem ruído quanto quando é usada SNR = 25 dB. O

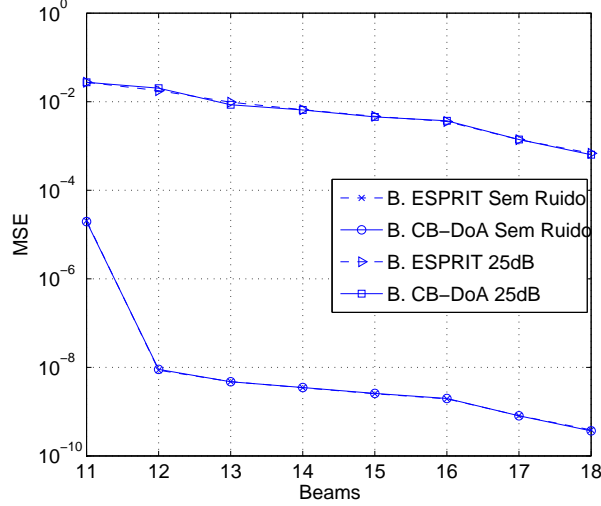


Figura 3.8: MSE medido para os algoritmos *BeamSpace* ESPRIT e *BeamSpace* CB-DoA, usando 2 fontes e 18 antenas de recepção, em função do número de *beams*.

MSE da versão *beamSpace* pode ocorrer devido a características inerentes ao próprio algoritmo ou por se utilizar uma banda inadequada no domínio DFT.

### 3.4.2 Comparações com o CRLB

Num segundo cenário de simulação, foi medido o MSE para os algoritmos *BeamSpace* ESPRIT e *BeamSpace* CB-DoA para 2 fontes e 18 antenas de recepção, usando-se um *beamSpace* de dimensão 12. Foi incluída também uma aproximação para o limite teórico CRLB, de acordo com a expressão simplificada apresentada na equação (3.30),

$$\text{CRLB} = \frac{6\sigma_N^2}{K(N^2 - 1)N\sigma_s^2}, \quad (3.43)$$

onde  $K = 500$  representa o número de *snapshots* e  $\sigma_s^2$  representa a potência das fontes. O resultado é bastante coerente com o que foi visto na subseção 3.3, com uma distância grande entre o *BeamSpace* CB-DoA e o CRLB. Essa distância já aparecia na versão *elementsSpace* e o processamento no *beamSpace* não reduz o MSE, já que o foco está apenas na diminuição da complexidade computacional.

Os resultados comparativos para o MSE das versões *beamSpace* do ESPRIT e CB-DoA eram até certa forma esperados, pelos resultados já obtidos para o *elementsSpace*, que pode ser visto como um caso específicos do *beamSpace*, e por serem feitas as mesmas operações em ambos os algoritmos para a redução de complexidade.

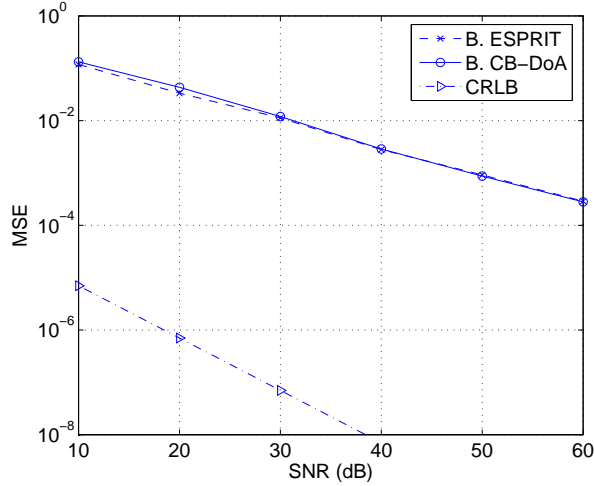


Figura 3.9: MSE para os algoritmos *Beamspace* ESPRIT e *Beamspace* CB-DoA, para  $M = 2$  fontes e  $N = 18$  antenas de recepção, em função da SNR, para 12 *beams*. CRLB também está incluído. O uso de 18 antenas neste cenário corresponde às versões *elementspace* de cada algoritmo.

### 3.4.3 Comparações entre *Beamspace* CB-DoA e *Elementspace* CB-DoA

A comparação entre dois algoritmos propostos, as versões *beamspace* e *elementspace* do algoritmo CB-DoA pode ser feita levando-se em conta vários fatores. Um dos mais importantes é a plataforma de *hardware* na qual os algoritmos são testados, já que qualquer estrutura paralela oferece vantagens para o algoritmo *beamspace* CB-DoA. Outro fator que influencia a comparação é a existência de um conhecimento prévio da localização das fontes, pois o processamento de algumas sub-bandas pode se revelar desnecessário. O número de sub-bandas utilizadas modifica a análise de complexidade do algoritmo.

Quando se compara a computação de uma sub-banda do algoritmo *Beamspace* CB-DoA com sua versão *elementspace*, a quantidade total de operações matriciais exigidas pela versão *beamspace* é maior. No entanto, como as operações no *beamspace* são realizadas sobre estruturas de matrizes e vetores de dimensões menores, então, há uma quantidade menor de operações escalares. Por exemplo, o algoritmo *Beamspace* CB-DoA requer 8 multiplicações matriciais, enquanto a versão *elementspace* exige apenas 3. No entanto, há uma redução no número de operações escalares, devido às

dimensões reduzidas das estruturas, o que pode compensar o aumento no número de operações escalares, visto que a complexidade assintótica da multiplicação entre matrizes é  $\mathcal{O}(n^3)$  [27].

Em relação à complexidade computacional, outra diferença entre os modelos *elementspace* e *beamspace* é a escolha dos *sub-arrays* com e sem sobreposição de antenas. Quanto maior o grau de sobreposição, obrigatória para a versão *beamspace*, maior é o conjunto de dados para se processar. Até o momento a versão trabalhada para o *elementspace* contém dados sem sobreposição, ou seja de mínima complexidade. O uso de sobreposição entre *sub-arrays* exigiria um número ainda maior de operações da versão *elementspace* na comparação com o *Beamspace* CB-DoA.

### 3.5 *Real* CB-DoA

Nesta seção será apresentado o algoritmo *Real* CB-DoA, uma alternativa que diminui a complexidade computacional do algoritmo proposto CB-DoA, com base na utilização de estruturas reais. A utilização de estruturas com elementos complexos é uma das características do algoritmo CB-DoA. Para se tornar uma alternativa de mais baixa complexidade ao algoritmo CB-DoA, foi necessário impor novas restrições à geometria das antenas de recepção. Essas novas restrições tornam o *array* de recepção como sendo do tipo CH, conforme foi apresentado na Subseção 2.3.5. A Figura 2.9 ilustra uma interpretação geométrica dessas restrições, isto é, antenas com posição simétrica em relação ao centróide do *array* devem possuir características semelhantes de recepção.

Na Subseção 2.3.5 foi mostrado que a equação de invariância original do ESPRIT (a mesma do CB-DoA) possui uma forma alternativa apresentada na equação (2.118), apenas com elementos reais. Foi mostrado ainda que a solução TLS para a equação de invariância puramente real possui uma equivalência com a solução TLS da equação de invariância original, conforme indicado na equação (2.137). A seguir serão feitas algumas operações matemáticas sobre a estrutura do modelo apresentado na Subseção 2.3.5 de modo a ser possível uma manipulação das estruturas de covariância apenas com elementos reais.

**Estruturas utilizadas** Considere a equação (2.128), aqui reproduzida,

$$\mathcal{E}(\mathbf{C}_{CH}) = \mathcal{T}(\mathbf{C}_X), \quad (3.44)$$

onde  $\mathcal{E}(\cdot)$  é dado pela equação (2.104), repetida a seguir por conveniência,

$$\mathcal{E}(\mathbf{M}) = \begin{bmatrix} \mathbf{M} & \mathbf{\Pi}_a \mathbf{M}^* \mathbf{\Pi}_a \end{bmatrix}. \quad (3.45)$$

Por sua vez,  $\mathcal{T}(\cdot)$  é definido a partir da equação (2.105), isto é,

$$\mathcal{T}(\mathbf{M}) = \psi(\mathcal{E}(\mathbf{M})). \quad (3.46)$$

Então realiza-se a transformação  $\mathcal{T}$  sobre  $\mathbf{C}_X$ ,

$$\mathcal{T}(\mathbf{C}_X) = \mathbf{Q}_N^H \begin{bmatrix} \mathbf{C}_X & \mathbf{C}_Y \end{bmatrix} \begin{bmatrix} \mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mathbf{\Pi}_K \end{bmatrix} \mathbf{Q}_{2K}. \quad (3.47)$$

Voltando à matriz de autovetores de  $\mathbf{R}$ , dada pela equação (2.115), separa-se em  $\mathbf{U}_{RS}$  os autovetores correspondentes aos  $M$  maiores autovalores de  $\mathbf{R}$ . Então  $\mathcal{T}(\mathbf{C}_X)$  pode ser reescrita como [1]

$$\mathcal{T}(\mathbf{C}_X) = \frac{1}{\sqrt{2}} \mathbf{Q}_N^H \begin{bmatrix} \mathbf{J}_X \mathbf{Q}_{2N} \mathbf{U}_{RS} & \mathbf{J}_Y \mathbf{Q}_{2N} \mathbf{U}_{RS} \end{bmatrix} \begin{bmatrix} \mathbf{I}_K & j\mathbf{I}_K \\ \mathbf{I}_K & -j\mathbf{I}_K \end{bmatrix}. \quad (3.48)$$

Definem-se mais uma vez as matrizes auxiliares de seleção

$$\mathbf{K}_X = \mathbf{Q}_N^H (\mathbf{J}_X + \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{2N}) \mathbf{Q}_{2N}, \quad (3.49)$$

$$\mathbf{K}_Y = j \mathbf{Q}_N^H (\mathbf{J}_X - \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{2N}) \mathbf{Q}_{2N}. \quad (3.50)$$

Utilizando o fato de que  $\mathbf{J}_X = \mathbf{\Pi}_N \mathbf{J}_Y \mathbf{\Pi}_{2N}$ , então

$$\mathcal{T}(\mathbf{C}_X) = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{K}_X \mathbf{U}_{RS} & \mathbf{K}_Y \mathbf{U}_{RS} \end{bmatrix}. \quad (3.51)$$

A partir do que foi visto na Subseção 2.3.5, há uma invariância na estrutura de  $\mathbf{U}_{ZZ}$ , matriz dos autovetores da covariância referente à matriz de *snapshots*  $\mathbf{Z}(i)$ . Tal invariância é mantida pelas novas matrizes de seleção  $\mathbf{K}_X$  e  $\mathbf{K}_Y$  [1] [37]. Com isso, chega-se a uma nova equação de invariância, já mostrada na equação (2.143), repetida aqui por conveniência,

$$\mathbf{K}_X \mathbf{U}_{RS} \mathbf{\Upsilon} = \mathbf{K}_Y \mathbf{U}_{RS}. \quad (3.52)$$

**Estimação** São então definidas duas matrizes de covariância geradas aplicando-se as novas matrizes de seleção  $\mathbf{K}_X$  e  $\mathbf{K}_Y$  sobre  $\mathbf{R}$ .

$$\mathbf{R}_{00} = \mathbf{K}_X(\mathbf{R} - \hat{\sigma}^2\mathbf{I})\mathbf{K}_X^H, \quad (3.53)$$

$$\mathbf{R}_{10} = \mathbf{K}_Y(\mathbf{R} - \hat{\sigma}^2\mathbf{I})\mathbf{K}_X^H, \quad (3.54)$$

onde  $\hat{\sigma}^2$  representa a potência estimada do ruído. Então, as equações (3.53) e (3.54) podem ser reescritas como

$$\mathbf{R}_{00} = \mathbf{K}_X\mathbf{Q}_{2N}^H(\mathbf{Z}(i)\mathbf{Z}^H(i) - \hat{\sigma}^2\mathbf{I})\mathbf{Q}_{2N}\mathbf{K}_X^H, \quad (3.55)$$

$$\mathbf{R}_{10} = \mathbf{K}_Y\mathbf{Q}_{2N}^H(\mathbf{Z}(i)\mathbf{Z}^H(i) - \hat{\sigma}^2\mathbf{I})\mathbf{Q}_{2N}\mathbf{K}_X^H, \quad (3.56)$$

São então realizadas as duas primeiras etapas de estimação SSI, de acordo com o apresentado na Subseção 2.3.1,

$$[\mathbf{U}_{00s}, \mathbf{\Lambda}_{00s}^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_{00}, \text{SSI1}(\mathcal{T}(\mathbf{C}_X))), \quad (3.57)$$

Define-se a matriz  $\mathbf{S}$  contendo os dados transmitidos

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}(0) & \mathbf{s}(1) & \dots & \mathbf{s}(K-1) \end{bmatrix}. \quad (3.58)$$

Então, existe uma matriz  $\mathbf{H}$  de posto completo tal que

$$\mathbf{K}_X\mathbf{Q}_{2N}^H\mathbf{S} = \mathbf{U}_{00s}\mathbf{\Lambda}_{00s}\mathbf{H}. \quad (3.59)$$

Utilizando-se a equação (2.143), que representa a equação de invariância, tem-se

$$\mathbf{K}_Y\mathbf{Q}_{2N}^H\mathbf{S} = \mathbf{K}_X\mathbf{Q}_{2N}^H\mathbf{S}\mathbf{\Upsilon} = \mathbf{U}_{00s}\mathbf{\Lambda}_{00s}\mathbf{H}\mathbf{\Upsilon}. \quad (3.60)$$

Mais uma vez, utilizando-se a equação de invariância,

$$\mathbf{R}_F = \mathbf{\Lambda}_{00s}^{-1}\mathbf{U}_{00s}^H\mathbf{K}_X\mathbf{Q}_{2N}^H\mathbf{S}\mathbf{\Upsilon}\mathbf{S}^H\mathbf{Q}_{2N}\mathbf{K}_X^H\mathbf{U}_{00s}\mathbf{\Lambda}_{00s}. \quad (3.61)$$

Realiza-se então uma operação TB, conforme definida na subseção 2.2.2,

$$[\mathbf{R}_F] = \text{TB}(\mathbf{U}_{00s}, \mathbf{\Lambda}_{00s}, \mathbf{R}_{10}). \quad (3.62)$$

A partir da equação (3.61) e da definição de TB, pode-se afirmar que

$$\mathbf{R}_F = \mathbf{H}\mathbf{\Upsilon}\mathbf{H}^H. \quad (3.63)$$

Utilizando-se a equação (2.138), ou seja,  $\mathbf{Y}_P = \mathbf{T}_P \mathbf{\Omega} \mathbf{T}_P^{-1}$ ,

$$\mathbf{R}_F = \mathbf{H} \mathbf{T}_P \mathbf{\Omega} \mathbf{T}_P^{-1} \mathbf{H}^H, \quad (3.64)$$

isto é, a matriz diagonal  $\mathbf{\Omega}$  tem seus elementos correspondendo aos autovalores de  $\mathbf{R}_F$ .

Por fim, estima-se os valores de DoA através da equação (2.139), reproduzida a seguir

$$\hat{\mathbf{\Phi}} = -(\mathbf{\Omega} - j\mathbf{I}_M)(\mathbf{\Omega} + j\mathbf{I}_M)^{-1}. \quad (3.65)$$

Por fim, é aplicada a etapa 4 de uma estimação SSI para DoA, conforme definido pela equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (3.66)$$

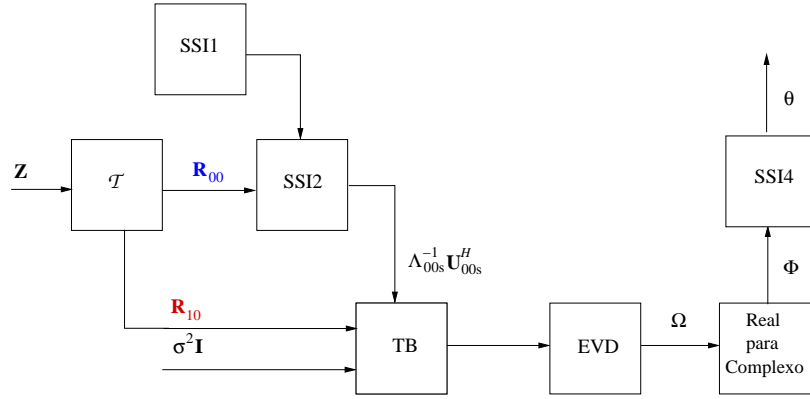


Figura 3.10: Diagrama de blocos representando o processo de estimação do algoritmo *Real* CB-DoA.

A Figura 3.10 mostra um resumo das operações do algoritmo *Real* CB-DoA.

### 3.5.1 Comparações entre *Real* CB-DoA e *Unitary* ESPRIT

**Complexidade Computacional:** Tanto o algoritmo *Unitary* ESPRIT, mostrado na subseção 2.3.5, quanto o algoritmo *Real* CB-DoA utilizam apenas operações reais. Nesta subseção deseja-se comparar ambos os algoritmos primeiramente em termos de complexidade computacional. Um resumo comparativo de ambos os algoritmos é encontrado na Tabela 3.7.

A partir dessa comparação, percebe-se que o algoritmo *Unitary* ESPRIT faz as seguintes operações:



Tabela 3.7: Descrição resumida dos algoritmos *Unitary* ESPRIT e *Real* CB-DoA.

<i>Unitary</i> ESPRIT	<i>Real</i> CB-DoA
$\Xi = \Pi_N \mathbf{J}_0 \Pi_{2N}$	$\Xi = \Pi_N \mathbf{J}_0 \Pi_{2N}$
$\mathbf{K}_0 = \mathbf{Q}_N^H (\mathbf{J}_0 + \Xi) \mathbf{Q}_{2K}$	$\mathcal{E}_Q(\mathbf{C}_X) = \mathcal{E}(\mathbf{C}_X) \mathbf{Q}_{2K}$
$\mathbf{K}_1 = \mathbf{Q}_N^H (\mathbf{J}_0 - \Xi) \mathbf{Q}_{2K}$	$\mathcal{T}_0(\mathbf{C}_X) = \mathbf{Q}_N^H (\mathbf{J}_0 + \Xi) \mathcal{E}_Q(\mathbf{C}_X)$
$\mathcal{T}(\mathbf{C}_X) = \mathbf{Q}_{2N}^H \mathcal{E}(\mathbf{C}_X) \mathbf{Q}_{2K}$	$\mathcal{T}_1(\mathbf{C}_X) = \mathbf{Q}_N^H j (\mathbf{J}_0 - \Xi) \mathcal{E}_Q(\mathbf{C}_X)$
$\mathbf{R} = \mathcal{T}(\mathbf{C}_X) \mathcal{T}^H(\mathbf{C}_X)$	$\mathbf{R}_{00} = \mathcal{T}_0(\mathbf{C}_X) \mathcal{T}_0^H(\mathbf{C}_X)$
$[\mathbf{U}_{\mathbf{R}_s}, \Lambda_{\mathbf{R}_s}^2] = \text{EVD}(\mathbf{R})$	$[\mathbf{U}_{00,s}, \Lambda_{00,s}^2] = \text{EVD}(\mathbf{R}_{00})$
$\mathbf{E}_x = \mathbf{K}_0 \mathbf{U}_{\mathbf{R}_s}, \mathbf{E}_y = \mathbf{K}_1 \mathbf{U}_{\mathbf{R}_s}$	
$\mathbf{E}_{\text{TLS}} = \begin{bmatrix} \mathbf{E}_x & \mathbf{E}_y \end{bmatrix}^H \begin{bmatrix} \mathbf{E}_x & \mathbf{E}_y \end{bmatrix}$	
$[\mathbf{U}_{\mathbf{E}_{\text{TLS}}}, \Lambda] = \text{EVD}(\mathbf{E}_{\text{TLS}})$	$\mathbf{F} = \Lambda_{00,s}^{-1} \mathbf{U}_{00,s}^H$
$\begin{bmatrix} \mathbf{E}_{00} & \mathbf{E}_{01} \\ \mathbf{E}_{10} & \mathbf{E}_{11} \end{bmatrix} = \mathbf{U}_{\mathbf{E}_{\text{TLS}}}$	$\mathbf{R}_{10} = \mathcal{T}_1(\mathbf{C}_X) \mathcal{T}_0(\mathbf{C}_X)^H - \hat{\sigma}^2 \mathbf{I}$
$\Upsilon = -\mathbf{E}_{01} \mathbf{E}_{11}^{-1}$	$\mathbf{R}_{\mathbf{F}} = \mathbf{F} \mathbf{R}_{10} \mathbf{F}^H$
$[\mathbf{U}_{\Upsilon}, \Omega] = \text{EVD}(\Upsilon)$	$[\mathbf{H} \mathbf{U}_{\Upsilon}, \Omega] = \text{EVD}(\mathbf{R}_{\mathbf{F}})$
$\hat{\Phi} = -(\Omega - j\mathbf{I}_M)(\Omega + j\mathbf{I}_M)^{-1}$	$\hat{\Phi} = -(\Omega - j\mathbf{I}_M)(\Omega + j\mathbf{I}_M)^{-1}$

- 3 EVDs (1 para uma matriz não-hermitiana de dimensões  $M \times M$ , 1 para uma matriz hermitiana de dimensões  $2N \times 2N$  e 1 para uma matriz hermitiana de dimensões  $2M \times 2M$ );
- 1 inversão para uma matriz de dimensões  $M \times M$ ;
- 16 multiplicações de matrizes (4 para produtos entre uma matriz com dimensões  $M \times N$  e uma matriz com dimensões  $N \times M$ , 2 para um produto de uma matriz de dimensões  $M \times 2N$  e uma matriz de dimensões  $2N \times N$ , 1 para um produto de um par de matrizes de dimensões  $M \times M$ , 1 para um produto de uma matriz de dimensões  $N \times N$  e uma matriz de dimensões  $N \times 2N$ , 1 para um produto de uma matriz de dimensões  $N \times 2N$  e uma matriz de dimensões  $2N \times 2N$ , 2 para produtos entre uma matriz de dimensões  $N \times N$  e uma matriz com dimensões  $N \times 2K$ , 2 para produtos entre uma matriz de dimensões  $N \times 2K$  e uma matriz com dimensões  $2K \times 2K$ , 1 para um produto

de uma matriz de dimensões  $2N \times 2N$  e uma matriz com dimensões  $2N \times 2K$ , e 1 para um produto de uma matriz de dimensões  $2N \times 2K$  e uma matriz com dimensões  $2K \times 2K$ ).

Já o algoritmo *Real* CB-DoA utiliza as seguintes operações:

- 2 EVDs (uma para uma matriz não-hermitiana de dimensões  $M \times M$  e 1 para uma matriz hermitiana de dimensões  $M \times M$ );
- 1 inversão de uma matriz diagonal de dimensões  $M \times M$ ;
- 12 multiplicações de matrizes (1 para um produto de uma matriz de dimensões  $M \times N$  e uma matriz de dimensões  $N \times N$ , 1 para um produto de uma matriz de dimensões  $M \times N$  e uma matriz de dimensões  $N \times M$ , 1 para um produto de uma matriz de dimensões  $M \times M$  e uma matriz de dimensões  $M \times N$ , 2 para um produto entre uma matriz com dimensões  $N \times N$  e uma matriz com dimensões  $N \times 2N$ , 1 para um produto de uma matriz de dimensões  $N \times 2N$  e uma matriz de dimensões  $2N \times 2N$ , 2 para produtos entre uma matriz com dimensões  $N \times 2K$  e uma matriz com dimensões  $2K \times N$ , 2 para produtos entre uma matriz de dimensões  $N \times 2N$  e uma matriz com dimensões  $2N \times 2K$ , e 1 para um produto de uma matriz de dimensões  $2N \times 2K$  e uma matriz de dimensões  $2K \times 2K$ ).

Observa-se que o número  $K$  de amostras é geralmente muito maior que  $M$  e  $N$ , que representam o número de fontes e a quantidade de antenas, respectivamente. Com isso, os termos envolvendo  $K^2$  tendem a dominar as expressões de complexidade computacional em cada algoritmo. Observando-se a Tabela 3.8, nota-se que, assintoticamente, o termo  $12NK^2$  predomina no algoritmo *Unitary* ESPRIT, enquanto o termo  $4NK^2$  predomina no algoritmo *Real* CB-DoA, ou seja, o algoritmo *Real* CB-DoA tende a apresentar 1/3 da complexidade do *Unitary* ESPRIT, para um valor grande de  $K$ .

**Desempenho:** Foram feitas simulações a fim de se verificar o desempenho dos algoritmos *Unitary* ESPRIT e *Real* CB-DoA em relação à métrica MSE. As simulações estão mostradas na Figura 3.11, utilizando inicialmente  $M = 2$  fontes e  $N = 12$  antenas de recepção e depois uma configuração com  $M = 4$  fontes e  $N = 12$  antenas.

Tabela 3.8: Comparação dos algoritmos *Unitary* ESPRIT e *Real* CB-DoA em termos do número de operações.

Operações [27]	<i>Unitary</i> ESPRIT		<i>Real</i> CB-DoA	
	#	Flops	#	Flops
EVD Matr. não-Hermitianas	1	$25M^3$	1	$25M^3$
EVD Matr. Hermitianas	2	$4N^2 + 4M^2$	1	$M^2$
Inversão	1	$(25/3)M^3$	–	–
Inversão Matr. Diagonais	–	–	1	$M$
Multiplicação	16	$4M^2N + 4MN^2 + M^3 +$ $18N^3 + 12N^2K + 12NK^2$	12	$4N^2K + 2M^2N + MN^2 +$ $10N^3 + 4NK^2$

Foram recebidos  $K = 7500$  *snapshots*. A métrica MSE foi calculada como média de 300 execuções de Monte Carlo. Tanto o *array manifold* quanto os dados transmitidos foram gerados aleatoriamente, mas os valores para o *array manifold* sofreram as restrições de geometria referentes à invariância à rotação e também restrições do tipo CH, como apresentadas na subseção 2.3.5. Resultados indicam grande similaridade em termos de MSE para os dois algoritmos. Em termos de complexidade computacional, como foi visto, o algoritmo *Real* CB-DoA tem aproximadamente 1/3 da complexidade computacional em relação ao algoritmo *Unitary* ESPRIT quando  $K$  é muito grande.

### 3.6 *Direct-Data* DoA (DD-DoA)

**Ambiente:** O algoritmo ESPRIT pode ser utilizado a partir de estruturas de covariância, como os algoritmos até aqui apresentados. No entanto, o ESPRIT também pode ser utilizado operando-se diretamente sobre os dados recebidos, substituindo-se as operações de EVD por operações de SVD [14], resultando em algo como um DD-ESPRIT. Esta seção tem como objetivo apresentar um algoritmo proposto para estimação possuindo baixa complexidade e que opera diretamente entre os dados recebidos, não necessitando o uso de estruturas de covariância.

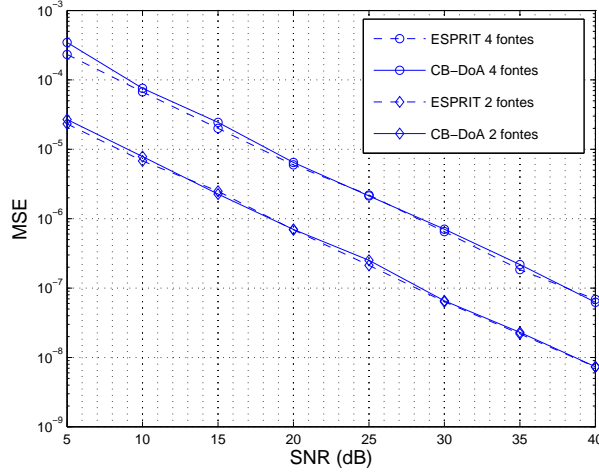


Figura 3.11: Comparação em termos de MSE para os algoritmos *Unitary* ESPRIT e *Real* CB-DoA em dois cenários de simulações, utilizando 2 e 4 fontes e um *array* de recepção de 12 antenas, em ambos os cenários.

Considere a modelagem de sistema dada pelas equações (2.52) e (2.53), repetida a seguir por conveniência,

$$\mathbf{x}(k) = \mathbf{A}_X \mathbf{s}(k) + \mathbf{n}_x(k), \quad (3.67)$$

$$\mathbf{y}(k) = \mathbf{A}_X \Phi \mathbf{s}(k) + \mathbf{n}_y(k). \quad (3.68)$$

No sistema há  $M$  fontes e  $N$  antenas de recepção. Sendo  $K$  o número de *snapshots* transmitidos, são definidas novas estruturas auxiliares

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}(0) & \mathbf{s}(1) & \dots & \mathbf{s}(K-1) \end{bmatrix}, \quad (3.69)$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(0) & \mathbf{x}(1) & \dots & \mathbf{x}(K-1) \end{bmatrix}, \quad (3.70)$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}(0) & \mathbf{y}(1) & \dots & \mathbf{y}(K-1) \end{bmatrix}, \quad (3.71)$$

$$\mathbf{N}_X = \begin{bmatrix} \mathbf{n}_x(0) & \mathbf{n}_x(1) & \dots & \mathbf{n}_x(K-1) \end{bmatrix}, \quad (3.72)$$

$$\mathbf{N}_Y = \begin{bmatrix} \mathbf{n}_y(0) & \mathbf{n}_y(1) & \dots & \mathbf{n}_y(K-1) \end{bmatrix}. \quad (3.73)$$

Com isso, o sistema pode ser descrito por

$$\mathbf{X} = \mathbf{A}_X \mathbf{S} + \mathbf{N}_X, \quad (3.74)$$

$$\mathbf{W} = \mathbf{X} + \mathbf{Y} = \mathbf{A}_X (\mathbf{I} + \Phi) \mathbf{S} + \mathbf{N}_X + \mathbf{N}_Y. \quad (3.75)$$

Supõe-se que a matriz  $\Phi$  de invariância tenha posto completo, e que o ruído seja modelado como branco tanto espacialmente quanto temporalmente.

**Estimação:** O processo de estimação dos métodos anteriores, com base em covariância, é feito utilizando-se a matriz  $\mathbf{R}_x$  de autocovariância, e  $\mathbf{R}_{xy}$ , de covariância cruzada. Quando o processamento é feito sobre os dados diretamente, a matriz  $\mathbf{X}$  assume o papel da autocovariância  $\mathbf{R}_x$ , no entanto, não há estrutura que assumo o papel da covariância cruzada  $\mathbf{R}_{xy}$ . No lugar de  $\mathbf{R}_{xy}$ , será usada a estrutura  $\mathbf{W} = \mathbf{X} + \mathbf{Y}$ , apresentada na equação (3.75).

Primeiramente, faz-se uma SVD sobre  $\mathbf{X}$ , como mostrado a seguir

$$\mathbf{X} = \mathbf{U}_X \mathbf{\Sigma}_X \mathbf{V}_X^H. \quad (3.76)$$

A seguir, a partir da matriz  $\mathbf{\Sigma}_X$ , são separados os  $M$  maiores valores singulares formando o subespaço dos sinais. É gerada a matriz  $\mathbf{\Sigma}_{X,S}$  contendo os valores singulares do subespaço dos sinais. Os vetores singulares à esquerda associados a esses valores singulares geram as colunas da matriz  $\mathbf{U}_{X,S}$ . Subtrai-se de  $\mathbf{\Sigma}_{X,S}$  uma estimativa da potência do ruído em sua diagonal, gerando a matriz  $\mathbf{\Sigma}_{X,(S-N)}$ . Tal processo de estimação é uma variação da função SSI2b mostrada na Subseção 2.3.1, ou seja, o equivalente às etapas 1 e 2 de estimação SSI está finalizado até este ponto do algoritmo.

Assim como nos algoritmos anteriores, é gerada a matriz  $\mathbf{F}$ , definida a seguir,

$$\mathbf{F}^{-1} = \mathbf{U}_{X,S} \mathbf{\Sigma}_{X,(S-N)}^{1/2}. \quad (3.77)$$

Faz-se então uma SVD sobre a matriz  $\mathbf{F}\mathbf{W}$ , que corresponde à etapa TB dos métodos propostos anteriores,

$$\mathbf{\Psi} = \text{SVD}(\mathbf{F}\mathbf{W}) = \mathbf{T}_1((1 + \alpha)\mathbf{I} + \mathbf{\Phi})\mathbf{T}_2^H, \quad (3.78)$$

onde  $\alpha$  contém uma parcela do ruído  $\mathbf{N}_X + \mathbf{N}_Y$ , presente na estrutura  $\mathbf{W}$  após passar pela projeção  $\mathbf{F}$ , ou seja, o ruído referente à parcela  $\mathbf{F}(\mathbf{N}_X + \mathbf{N}_Y)$ . Com isso, uma estimação  $\hat{\mathbf{\Phi}}$  é obtida a partir dos valores singulares de  $\mathbf{\Psi}$ .

Após a estimação dos  $\hat{\phi}$ , elementos da matriz  $\hat{\mathbf{\Phi}}$ , os DoAs são estimados a partir da equação (2.54), correspondente à quarta etapa de estimação SSI no cenário de estimação de DoA,

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (3.79)$$

Na Figura 3.12 está representado um diagrama de blocos resumindo as operações do algoritmo DD-DoA.

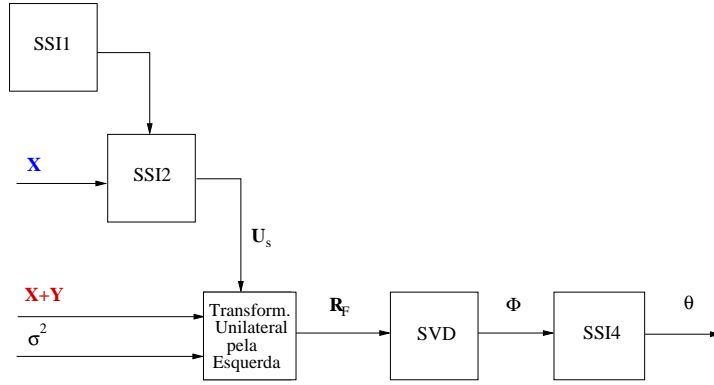


Figura 3.12: Diagrama de blocos representando o processo de estimação do algoritmo DD-DoA.

### 3.6.1 Comparações entre DD-DoA e TLS-ESPRIT

**Complexidade Computacional:** Serão feitas algumas comparações entre o algoritmo proposto DD-DoA e a versão SVD do algoritmo TLS-ESPRIT. Uma descrição resumida dos dois algoritmos é encontrada na Tabela 3.9, onde se utilizou a notação  $\mathbf{Z} = [\mathbf{X}^T \quad \mathbf{Y}^T]^T$ . Observa-se que, para o algoritmo DD-DoA, foi utilizada diretamente a estrutura  $\mathbf{W} = \mathbf{X} + \mathbf{Y}$ , não se necessitando processar  $\mathbf{Y}$ . Além disso, a matriz  $\mathbf{F}$  foi computada diretamente através de  $\Sigma_{X,S-N}^{-1} \mathbf{U}_{X,S}^H$ , o que corresponde a uma economia de operações, pois  $\Sigma_{X,S-N}$  é uma matriz diagonal, portanto, o cálculo de sua inversa é de baixa complexidade computacional.

O algoritmo ESPRIT requer

- 1 SVD de uma matriz de dimensões  $P \times K$ ;
- 1 EVD de uma matriz não-hermitiana de dimensões  $P \times P$ ;
- 1 inversão de uma matriz completa  $P \times P$ ;
- 5 multiplicações, todas entre matrizes de dimensões  $P \times P$ .

Por sua vez o algoritmo DD-DoA requer

- 2 SVD (1 para uma matriz de dimensões  $P \times P$  e uma de dimensões  $M \times K$ ;
- 2 multiplicações (uma entre uma matriz de dimensões  $P \times P$  e uma matriz de dimensões  $P \times K$  e uma entre uma matriz de dimensões  $M \times M$  e uma matriz de dimensões  $M \times P$ ).

Tabela 3.9: Descrição resumida dos algoritmos DD-DoA e TLS-ESPRIT, na versão que opera diretamente sobre os dados.

TLS-ESPRIT	DD-DoA
$[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^H] = \text{SVD}(\mathbf{Z})$	$\mathbf{X} = \mathbf{J}_X \mathbf{Z}$
$\mathbf{U}_0 = \mathbf{J}_X \mathbf{U}$	$\mathbf{W} = (\mathbf{J}_X + \mathbf{J}_Y) \mathbf{Z}$
$\mathbf{U}_1 = \mathbf{J}_Y \mathbf{U}$	$[\mathbf{U}_X, \mathbf{\Sigma}_X, \mathbf{V}_X^H] = \text{SVD}(\mathbf{X})$
$\mathbf{U}_a = \begin{bmatrix} \mathbf{U}_0^* \\ \mathbf{U}_1^* \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 & \mathbf{U}_1 \end{bmatrix}$	$\mathbf{F}^{-1} = \mathbf{\Sigma}_{X,(S-N)}^{-1/2} \mathbf{U}_{X,S}^H$
$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{U}_a)$	$[\mathbf{T}_1, (1+\alpha)\mathbf{I} + \hat{\mathbf{\Phi}}, \mathbf{T}_2] = \text{SVD}(\mathbf{F}^{-1} \mathbf{W})$
$\mathbf{E}_{12} = \mathbf{J}_X \mathbf{E} \mathbf{J}_Y^T$	
$\mathbf{E}_{22} = \mathbf{J}_Y \mathbf{E} \mathbf{J}_Y^T$	
$\mathbf{\Psi} = -\mathbf{E}_{12}^{-1} \mathbf{E}_{22}$	
$[\mathbf{T}_1, \hat{\mathbf{\Phi}}] = \text{EVD}(\mathbf{\Psi})$	

A Tabela 3.10 apresenta uma comparação entre TLS-ESPRIT, em sua versão que opera sobre os dados diretamente, e DD-DoA em termos do quantidade de operações de cada um dos algoritmos. Observa-se que, no caso de o número  $K$  de *snapshots* ser muito maior que o valor de  $M$  e  $P$ , o que é um cenário bastante comum, o termo dominante na complexidade computacional do TLS-ESPRIT é  $4PK^2$ , enquanto o termo dominante de complexidade do DD-DoA é  $2(M+P)K^2$ . Com isso, quando  $K$  é muito maior que  $M$  e  $P$ , DD-DoA apresenta uma redução de  $(M+P)/2P$  na complexidade computacional em comparação com o algoritmo TLS-ESPRIT.

Nota-se uma dependência um pouco maior desta comparação de complexidade em relação à implementação das operações, visto que as operação que domina a análise de complexidade é a SVD.

**Desempenho:** Foram realizados experimentos para comparar o algoritmo DD-DoA com o método TLS-ESPRIT, na versão que opera diretamente sobre os dados, em termos da métrica MSE, definida na equação (3.28). As simulações foram realizadas com  $M = 2$  fontes com ângulos  $-8^\circ$  e  $9^\circ$ , com 6000 *snapshots*. O parâmetro  $\alpha$  foi ajustado para aproximadamente  $10^{-2}$  vezes o menor valor singular de  $\mathbf{W}$ . A Figura 3.13 contém os resultados das simulações. Para  $M = 2$  fontes e  $P = 14$

Tabela 3.10: Comparação entre TLS-ESPRIT, em sua versão que opera diretamente sobre os dados, e DD-DoA em termos de número de operações com matrizes.

Operações [27]	TLS-ESPRIT		DD-DoA	
	#	Flops	#	Flops
SVD $\mathcal{O}(2mn^2)$	1	$4PK^2$	2	$2PK^2 + 2MK^2$
EVD $\mathcal{O}(25n^3)$	1	$200P^3$	–	–
EVD - Matr. Hermitianas $\mathcal{O}(n^2)$	1	$P^2$	–	–
Inversa de Matr. Completa $\mathcal{O}(n^3)$	1	$P^3$	–	–
Multiplicações $\mathcal{O}(mnk)$	5	$5P^3$	2	$P^2K + M^2P$
Total		$4PK^2 + 206P^3 + P^2$		$2PK^2 + 2MK^2 + P^2K + M^2P$

antenas, verificou-se que o algoritmo TLS-ESPRIT apresenta um menor MSE em comparação com o DD-DoA. Como visto nesta mesma seção, o algoritmo DD-DoA apresenta uma complexidade computacional reduzida em comparação com o TLS-ESPRIT. A razão entre as complexidades é de aproximadamente  $(M + P)/2P$ . No caso, há uma redução de aproximadamente 42% na quantidade de flops necessários. Então, para se verificar o quanto de redução de complexidade computacional se traduz em degradação de MSE, ainda na Figura 3.13, aparece a curva para o DD-DoA com  $M = 2$  fontes e  $P = 26$  antenas, ou seja, com a complexidade computacional dos algoritmos aproximadamente igualada. Verifica-se um menor MSE do DD-DoA em toda a faixa de SNRs. Ainda na Figura 3.13, é calculada a curva de MSE teórica para o DD-DoA com  $M = 2$  fontes e  $P = 26$  de acordo com a análise de MSE para o algoritmo DD-DoA mostrada no Apêndice A.

Verifica-se então que o algoritmo DD-DoA apresenta um compromisso da métrica MSE em relação a complexidade computacional quando comparado com o TLS-ESPRIT. É observado que, para uma configuração com mesmo número de antenas, há um menor MSE para o ESPRIT, mas a complexidade do DD-DoA é 42% menor. No entanto, quando a complexidade computacional de ambos os algoritmos é igualada, o algoritmo DD-DoA apresenta um menor MSE. Além disso, a curva teórica apresentada tem valores bastante próximos em relação aos simulados para o algoritmo DD-DoA.



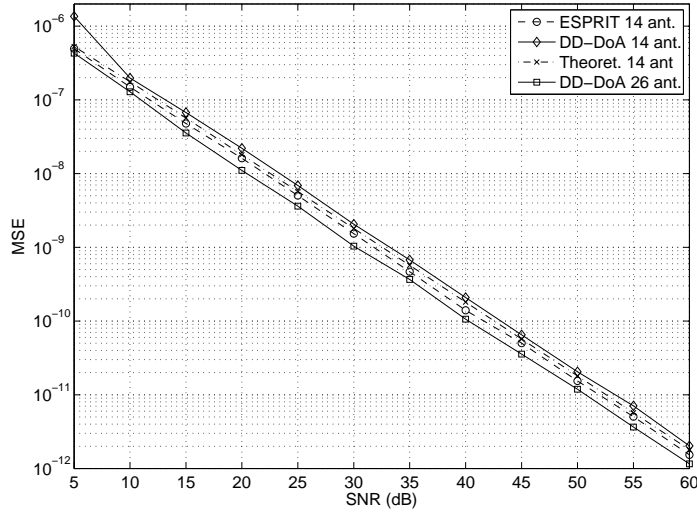


Figura 3.13: Comparação entre TLS-ESPRIT, na versão que opera diretamente sobre os dados, e DD-DoA em termos de MSE para  $M = 2$  fontes and  $P = 14$  antenas, e também para o DD-DoA com  $M = 2$  fontes e  $P = 26$  antenas. A curva para MSE teórico, calculado utilizando a equação (A.14), para  $M = 2$  fontes e  $P = 14$  antenas também pode ser vista.

### 3.6.2 Comparações entre DD-DoA e CRLB

Foram realizadas simulações para se comparar o algoritmo DD-DoA com a expressão simplificada do CRLB apresentada na equação (3.30), repetida mais uma vez por comodidade,

$$\text{CRLB} = \frac{6}{\text{SNR } P^3}. \quad (3.80)$$

Foram realizadas simulações para o cenário com  $M = 1$  fonte e  $P = 8$  antenas, mais uma vez para os ângulos de  $-8^\circ$  e  $9^\circ$ , comparando-se com o limite estatístico CRLB. Os resultados estão mostrados na Figura 3.14. Como a aproximação utilizada para o CRLB é válida para uma grande quantidade de *snapshots*, foram usados 18000 *snapshots* neste cenário de simulação. Percebe-se a diferença entre o MSE para o algoritmo DD-DoA e seu limite teórico, mostrando que é possível que se consiga melhorar o desempenho do algoritmo utilizado em termos de MSE.

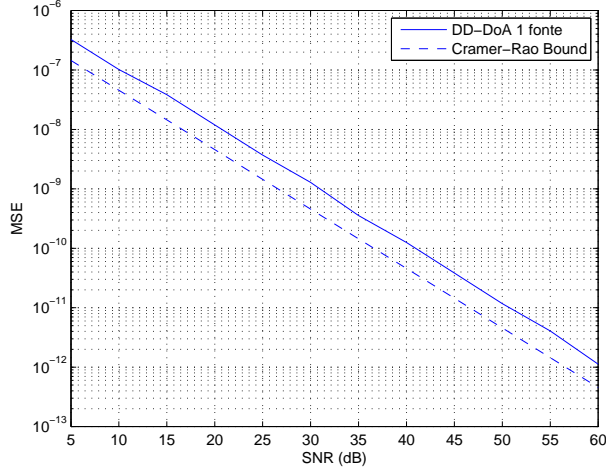


Figura 3.14: MSE para o algoritmo DD-DoA para  $M = 1$  fonte e  $P = 8$  antenas, e seu limite teórico CRLB.

## 3.7 Discussões do Capítulo

Esta subseção apresenta algumas discussões complementares aos tópicos abordados neste capítulo. Primeiramente é justificado o uso da versão TLS do ESPRIT como algoritmo a ser comparado com os algoritmos propostos. A seguir, alguns métodos cujas formulações têm alguma semelhança com os algoritmos propostos são comparados com o CB-DoA, mostrando as semelhanças e diferenças de conceitos envolvidos nas formulações de cada um deles. Por fim, são abordadas algumas questões internas do funcionamento dos algoritmos propostos.

### 3.7.1 Sobre as Comparações com o TLS-ESPRIT

**Solução TLS:** A formulação dos algoritmos propostos com regularização a partir da potência do ruído guarda uma particular relação com algoritmos do tipo TLS, definido na equação (2.38). Como mostrado em [35], a solução TLS para um sistema linear  $\mathbf{Ax} = \mathbf{b}$  é dada por

$$\hat{\mathbf{x}}_{\text{TLS}} = (\mathbf{R}_a - \sigma \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b}, \quad (3.81)$$

onde  $\mathbf{R}_a$  representa a correlação determinística  $\mathbf{A}^H \mathbf{A}$  e  $\sigma$  representa o menor valor singular de  $\begin{bmatrix} \mathbf{A} & \mathbf{b} \end{bmatrix}$ . Tal semelhança tanto nas estruturas quanto na forma de regularização em relação às estruturas processadas pelo CB-DoA reforça a escolha da versão TLS do ESPRIT nas comparações de desempenho do CB-DoA.

**Comparações com LS-ESPRIT:** Em todo este capítulo, as comparações do CB-DoA foram feitas com a versão TLS do ESPRIT, que é a mais popular e que apresenta uma relação com o CB-DoA como já mostrado nesta seção. A fim de se proporcionar uma comparação com a versão LS do ESPRIT, é apresentada a Tabela 3.11. Neste caso, percebe-se que a redução de complexidade proporcionada pelo CB-DoA não é tão grande quanto em comparação com o TLS-ESPRIT, mas ainda assim é bastante significativa. É mostrado em [47] que o LS-ESPRIT apresenta MSE ligeiramente maior que o TLS-ESPRIT quando se usam poucos *snapshots*, mas ambos possuem assintoticamente o mesmo desempenho em termos de MSE, ao se aumentar o número de amostras utilizadas.

Tabela 3.11: Comparação para as operações requeridas por LS-ESPRIT e CB-DoA.

Operação	Complexid. [27]	LS-ESPRIT	CB-DoA
EVD Matrizes Não-Hermitianas	$\mathcal{O}(25n^3)$	2	1
EVD Matrizes Hermitianas	$\mathcal{O}(n^2)$	–	1
Inversão Completa	$\mathcal{O}(2n^3/3)$	2	1
Inversão Matrizes Diagonais	$\mathcal{O}(n)$	–	1
Multiplicação	$\mathcal{O}(n^3)$	5	3
Subtração	$\mathcal{O}(n^2)$	–	1

### 3.7.2 Comparações com outros Algoritmos

**Relações com MP:** O algoritmo CB-DoA apresenta algumas relações com algoritmos do tipo MP, apresentados na Subseção 2.4, para *harmonic retrieval* em ambientes com ruído branco, como o algoritmo apresentado em [18]. Ao contrário dos métodos MP, porém, o algoritmo CB-DoA generaliza os resultados para um contexto no qual as antenas obedecem apenas as relações de invariância, não tendo obrigatoriamente um espaçamento uniforme. Com isso, o CB-DoA possui as mesmas restrições geométricas que o ESPRIT, o que justifica uma comparação direta entre ESPRIT e CB-DoA. Além disso, a modelagem utilizada para o ruído é mais genérica, sem obrigatoriedade de ser espacialmente decorrelacionado e os conjuntos

$X$  e  $Y$  de antenas de recepção podem ser inteiramente disjuntos no CB-DoA, do mesmo modo que no ESPRIT.

Outro algoritmo que utiliza apenas operações reais para estimar DoA está apresentado em [48], e se baseia em métodos do tipo MP, como o apresentado na Subseção 2.4. Esse método possui algumas operações, principalmente as iniciais, bastante semelhantes ao *Unitary* ESPRIT. Por exemplo, as seis primeiras linhas da Tabela 3.7 para o *Unitary* ESPRIT são também utilizadas pelo algoritmo descrito [48]. No entanto, esse algoritmo requer uma GSVD, uma operação bastante complexa computacionalmente, que não é utilizada pelo *Unitary* ESPRIT e *Real* CB-DoA, tornando o algoritmo baseado em MP mais complexos que esses métodos. Com isso, não foi usado em comparações de complexidade computacional neste capítulo.

Como também mencionado na Subseção 2.4, o método ESPRIT para *harmonic retrieval* apresentado em [38] utiliza as mesmas estruturas que o método CB-DoA, ou seja,  $\mathbf{R}_x$  e  $\mathbf{R}_{xy}$ . Esse método, no entanto, exige uma amostragem espacial uniforme dentro de um *snapshot*, caracterizando um *array manifold* equivalente ao de um ULA, e requer que o ruído seja branco, isto é, impondo restrições de sistemas mais rígidas que o CB-DoA. Em seu processamento, é feita uma GEVD no par  $\{\mathbf{R}_x, \mathbf{R}_{xy}\}$ , operação bastante complexa que não é exigida pelo CB-DoA em caso de ruído branco.

Os algoritmos CB-DoA, *Beam-space* CB-DoA e *Real* CB-DoA utilizam uma operação TB sobre o *pencil* ( $\mathbf{R}_{xy} - \sigma^2 \mathbf{R}_N$ ). Essa operação, apesar de ser redundante, resulta em uma EVD de uma matriz hermitiana no caso específico de os dados de entrada serem descorrelacionados, o que representa um ganho de complexidade nesse caso específico. Tal característica não é apresentada pelas diferentes versões do ESPRIT encontradas, e é herdada da formulação originalmente feita pelo algoritmo TXK.

**Relações com o Método de Zoltowski:** Outras modelagens MP como em [49] apresentam similaridades com CB-DoA. Em [49], são apresentados dois métodos de estimação de DoA. O primeiro deles tem uma modelagem próxima ao CB-DoA, mas trabalha com os dados diretamente, usando SVD ao invés de uma modelagem de covariância, num ambiente sem ruído e aproveita-se de propriedades de redução

de posto. Esse método é denominado *método das rotações simultâneas*. Outro algoritmo apresentado, já para um ambiente com ruído espacialmente branco, utiliza matrizes de covariância, mas adota uma abordagem bastante diferente do CB-DoA, usando conceitos como *problema de Procrustes* [50], que procura resolver o problema de estimação de DoA com base em SVD dos *sub-arrays*  $\mathbf{X}$  e  $\mathbf{Y}$ .

### 3.7.3 Sobre Questões Internas dos Algoritmos Propostos

**Proposição:** Na Subseção 2.2.2 foi apresentado um teorema de identificação de canal associado ao algoritmo TXK. Tenta-se agora enunciar uma proposição equivalente para os algoritmos CB-DoA e ESPRIT. No algoritmo TXK, a identificação é feita de tal maneira que se permita resolver uma ambiguidade de identificação causada por rotação em relação à matriz de TSRs do canal, ou seja, o algoritmo equaliza os dados mesmo que várias matrizes  $\mathbf{H}_i$  referentes a TSRs relacionadas por  $\mathbf{H}_{i1} = \mathbf{Q}\mathbf{H}_{i2}$ , onde  $\mathbf{Q}$  é uma matriz unitária, resultem na mesma sequência recebida.

Do mesmo modo, o algoritmo CB-DoA permite que se identifiquem os DoAs mesmo que a matriz de *array manifold*  $\mathbf{A}_i$  apresente em sua identificação uma ambiguidade tal que  $\mathbf{A}_{i1} = \mathbf{T}\mathbf{A}_{i2}$ , onde  $\mathbf{T}$  seja quadrada e de posto completo. Esta proposição representa o equivalente ao teorema de identificação, mas tendo como contexto a estimação de DoA. A prova da proposição é deixada como proposta para trabalhos futuros.

**Ruído Espacialmente Correlacionado:** Nos algoritmos descritos, o CB-DoA apresenta o ruído com uma matriz de covariância espacial possivelmente diferente da identidade, indicando uma possível correlação entre as interferências recebidas pelas antenas. Os demais algoritmos foram apresentados considerando o ruído espacialmente decorrelacionado. Essa escolha foi feita obedecendo-se a descrição das versões dos algoritmos ESPRIT equivalentes a cada um deles. Enquanto o *Elementspace* ESPRIT original foi descrito para um campo de ruído correlacionado espacialmente, os algoritmos *Beamspace* ESPRIT e *Unitary* ESPRIT foram descritos com campo de ruído espacialmente branco. Foi seguida a mesma descrição, mas é possível o uso dos algoritmos *Beamspace* CB-DoA e *Unitary* CB-DoA para ruído espacialmente correlacionado. O uso de ruídos correlacionados exige que operações

de GEVDs sejam usadas no lugar de EVDs, o que faz os algoritmos da família *Covariance-based* apresentarem uma complexidade computacional ainda menor em relação à do ESPRIT.

**Complexidade de Operações com Dados Reais** A partir do que foi visto nesta seção, percebe-se que o esforço em usar estruturas do tipo CH leva à utilização apenas de operações sobre dados reais. A partir da teoria exposta no livro [27], chega-se à comparação entre operações sobre dados reais e complexos como mostrado na Tabela 3.12. A operação de EVD sobre matrizes não-hermitianas não tem sua

Tabela 3.12: Comparação da quantidade de flops necessários em operações sobre dados reais e complexos, considerando-se a quantidade de operações reais.

Operação	Dados Reais	Dados Complexos
EVD Matrizes não-Hermitianas	$\mathcal{O}(25n^3)$	$\mathcal{O}(25n^3)$
EVD Matrizes Hermitianas	$\mathcal{O}(n^2)$	$\mathcal{O}(2n^2)$
Inversão Matrizes Diagonais	$\mathcal{O}(n)$	$\mathcal{O}(2n)$
Inversão Completa	$\mathcal{O}(2n^3/3)$	$\mathcal{O}(8n^3/3)$
Multiplicação	$\mathcal{O}(n^3)$	$\mathcal{O}(4n^3)$

complexidade mudada quando aplicada sobre dados reais ao se comparar com dados complexos. Uma matriz não-hermitiana com dados reais pode possuir autovalores e autovetores complexos. Com isso, o processamento interno da EVD sobre matrizes não-hermitianas com elementos reais demanda a utilização de estruturas complexas.

Ao comparar os dados da Tabela 3.12 com as análises de complexidade vistas no capítulo, percebe-se uma diferença em relação às expressões de complexidade assintótica para dados complexos. Isso ocorre pois a Tabela 3.12 mostra expressões de complexidade com base na quantidade de operações reais, enquanto nas demais, a expressão tinha como base o número de operações complexas.

## 3.8 Conclusões do Capítulo

Neste capítulo foram apresentados novos métodos de estimação paramétrica no cenários de estimação de DoA em ambientes com múltiplas antenas. Os algoritmos propostos apresentam complexidade computacional menor que a do método tradicional ESPRIT de estimação paramétrica em cada um dos cenários estudados, com MSE aproximadamente equivalente. Por fim, são feitas comparações dos métodos desenvolvidos com métodos existentes na literatura, que apresentam similaridades com o CB-DoA.

Os novos algoritmos propostos para estimação de DoA motivaram a extensão dos métodos para um novo tipo de cenário. O próximo capítulo aborda novos algoritmos propostos, dentro de um contexto de comunicações utilizando multiportadoras. No Apêndice A, podem-se encontrar deduções matemáticas para expressões teóricas de MSE para dois dos algoritmos vistos neste capítulo, CB-DoA e DD-DoA.

# Capítulo 4

## Algoritmos para Estimação de CFO

### 4.1 Introdução do Capítulo

Neste capítulo estão apresentadas a modelagem de um sistema de comunicações utilizando OFDM (*orthogonal frequency-division multiplexing*) bem como a versão ESPRIT para estimação de CFO (*carrier frequency offset*) em portadoras para sistemas OFDM. Além disso, são apresentados dois algoritmos propostos para estimação do CFO com complexidade reduzida em relação ao algoritmo ESPRIT:

- *CB-CFO (Covariance-Based Carrier Frequency Offset Estimation)*: alternativa de baixa complexidade em relação ao algoritmo ESPRIT para estimação de CFO;
- *Unitary ESPRIT* para estimação de CFO: versão do ESPRIT para estimação de CFO utilizando apenas operações reais.

### 4.2 CFO em Sistemas OFDM

Nesta seção será apresentado um outro ambiente no qual foram aplicados os algoritmos de estimação apresentados nesta tese. A aplicação almejada é a estimação de CFO em sistemas OFDM.



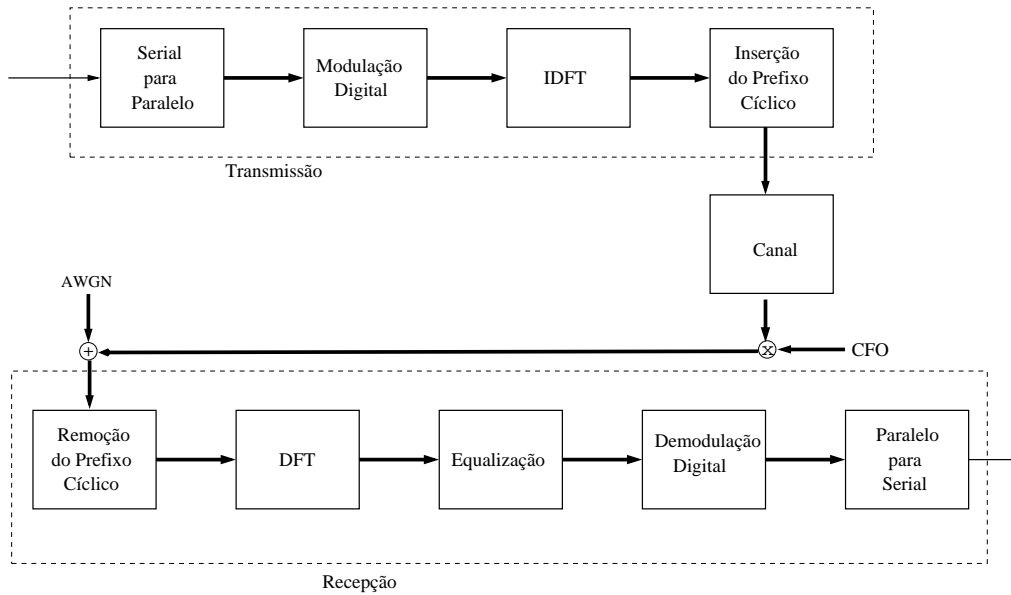


Figura 4.1: Sistema com multiplexação por portadoras, do tipo OFDM. O CFO é modelado como um ruído multiplicativo que interfere com o sinal na saída do canal.

#### 4.2.1 Descrição do Ambiente

Sistemas com multiplexação por portadoras são bastante usados no intuito de se combater os efeitos de ISI. Dentre as aplicações atualmente utilizadas que contêm multiplexação por portadoras podem-se citar padrões de televisão digital, como o DVB (*digital video broadcasting*), padrões de rádio digital, como o DAB (*digital audio broadcasting*), conexão à Internet utilizando DSL (*digital subscriber line*) [13], além do padrão WiMAX (*wireless interoperability for microwave access*), para transmissões em rede *wireless*.

Para garantir o combate efetivo à ISI, as múltiplas portadoras devem ser ortogonais entre si. Um dos sistemas com multiplexação por portadoras muito utilizados é o OFDM, em que as portadoras são moduladas por uma *inverse discrete Fourier transform* (IDFT) e demoduladas por uma DFT. Além disso, há a inserção de um prefixo contendo parte do próprio bloco a ser transmitido, gerando o chamado prefixo cíclico. O sistema OFDM é representado na Figura 4.1, onde o ruído multiplicativo de fase, causador do desvio na frequência da portadora, CFO, está explicitado. O funcionamento desse sistema é explicado mais detalhadamente em [13].

Em alguns sistemas OFDM, parte das portadoras não são usadas em cada bloco, estando presentes apenas para melhorar a qualidade na transmissão e para o combate

das interferências presentes. Essas portadoras são denominadas *virtuais*.

Considere um sistema OFDM com  $N$  portadoras, sendo  $P$  delas efetivamente usadas. As  $(N - P)$  portadoras restantes são portadoras virtuais. Define-se  $\mathbf{s}(k)$  um bloco de  $P$  dados a serem transmitidos. Além disso,  $\mathbf{W}$  é definida como uma matriz de IDFT de dimensões  $N \times N$ , sendo seu elemento na linha  $l$  e coluna  $c$  definido por

$$\{\mathbf{W}\}_{l,c} = \frac{1}{\sqrt{N}} e^{j \frac{2\pi lc}{N}} \quad (4.1)$$

Considere a matriz  $\mathbf{W}_P$  possuindo as  $P$  primeiras colunas de  $\mathbf{W}$ , ou seja, é uma matriz de dimensões  $N \times P$  correspondendo à IDFT das portadoras não-virtuais do sistema. Não se supõe a existência de portadoras piloto no sistema. Sendo  $\mathbf{x}(k)$  o conjunto de dados após ser transformado por uma IDFT [6], tem-se

$$\mathbf{x}(k) = \mathbf{W}_P \mathbf{s}(k). \quad (4.2)$$

Um prefixo cíclico de tamanho  $L$ , maior que o comprimento da resposta ao impulso do canal, deve ser concatenado ao bloco de dados  $\mathbf{x}(k)$ . Com o uso do prefixo cíclico, a matriz de convolução linear correspondente ao canal, que possui estrutura do tipo Toeplitz, torna-se também circulante, estrutura característica de uma convolução circular [13].

Considera-se  $\mathbf{H}$  a matriz circulante representativa do canal. Então a matriz IDFT  $\mathbf{W}$  de transmissão e sua contrapartida de recepção, a matriz  $\mathbf{W}^H$  correspondente à DFT, diagonalizam a matriz de canal de acordo com a seguinte equação [13]

$$\mathbf{H}_D = \mathbf{W}^H \mathbf{H} \mathbf{W}, \quad (4.3)$$

onde  $\mathbf{H}_D$  é uma matriz diagonal de dimensões  $N \times N$ .

Forma-se então a matriz auxiliar  $\mathbf{H}_{D,P}$  contendo as  $P$  primeiras colunas e as  $P$  primeiras linhas de  $\mathbf{H}_D$ . No receptor, após a remoção do prefixo cíclico o vetor recebido  $\mathbf{y}(k)$ , de comprimento  $P$ , está relacionado aos dados transmitidos  $\mathbf{s}(k)$  através da seguinte expressão

$$\mathbf{y}(k) = \mathbf{W}_P \mathbf{H}_{D,P} \mathbf{s}(k), \quad (4.4)$$

onde se considera, a princípio, que o sistema não possua AWGN. A equação (4.4) é uma modelagem do sistema no domínio do tempo, ou seja, antes de se realizar a DFT no receptor sobre os dados recebidos.

Com o canal diagonalizado, uma equalização ZF [17] torna-se computacionalmente simples, modelada pela matriz  $\mathbf{G}$ , com estrutura diagonal e com dimensões  $N \times N$ , que representa a equalização através de um conjunto de filtros, cada um com apenas um coeficiente. Forma-se então outra estrutura auxiliar, a matriz  $\mathbf{G}_P$  contendo apenas as  $P$  primeiras colunas e  $P$  primeiras linhas de  $\mathbf{G}$ . Com isso, para o caso ZF sem outras interferências no canal, é feita a estimação da sequência transmitida, da seguinte maneira,

$$\hat{\mathbf{s}}(k) = \mathbf{G}_P \mathbf{W}_P^H \mathbf{y}(k) = \mathbf{G}_P \mathbf{W}_P^H \mathbf{W}_P \mathbf{H}_D \mathbf{s}(k) = \mathbf{s}(k), \quad (4.5)$$

ou seja, no caso sem AWGN há uma estimação perfeita da sequência transmitida.

Supõe-se, no entanto, que haja um ruído de fase  $\theta$ , causando CFO no sistema OFDM. Além disso, considera-se que o ruído seja determinístico e cumulativo, tal que, para a  $i$ -ésima portadora do  $k$ -ésimo bloco, o CFO  $\delta(k, i)$  é modelado como

$$\delta(k, i) = e^{j((k-1)(N+L)+i)\theta}, \quad 0 \leq i < N. \quad (4.6)$$

Então, ao se levar em conta o efeito de CFO, a equação (4.4) é modificada para

$$\mathbf{y}(k) = \mathbf{E} \mathbf{W}_P \mathbf{H}_{D,PS}(k) e^{j(k-1)\theta(N+L)}, \quad (4.7)$$

onde  $\mathbf{E}$ , de dimensões  $N \times N$ , representa o efeito de CFO dentro de um mesmo bloco OFDM,

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & e^{j\theta} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{j(N-1)\theta} \end{bmatrix}. \quad (4.8)$$

Com isso, a nova estimativa é obtida a partir de

$$\hat{\mathbf{s}}(k) = \mathbf{G}_P \mathbf{W}_P^H \mathbf{E} \mathbf{W}_P \mathbf{H}_{D,PS}(k) e^{j(k-1)\theta(N+L)}. \quad (4.9)$$

Como a expressão  $\mathbf{W}_P^H \mathbf{E} \mathbf{W}_P$  não corresponde à matriz identidade e, também devido à presença do ruído multiplicativo  $e^{j(k-1)\theta(N+L)}$ , o uso de um equalizador ZF gera uma estimativa  $\hat{\mathbf{s}}(k)$  distorcida em relação a  $\mathbf{s}(k)$ , mesmo sem AWGN. A estratégia empregada nos algoritmos contidos nesta tese para eliminar o efeito do CFO consiste em estimar e compensar o efeito do ruído de fase  $\theta$  no receptor.

A Tabela 4.1 mostra resumidamente alguns outros efeitos causados pelo CFO sobre um sistema OFDM. As interferências IBI (*interblock interference*) e ICI (*intercarrier interference*) estão relacionadas a interferências mútuas entre blocos transmitidos, e entre portadoras de um mesmo bloco, respectivamente.

Tabela 4.1: Tabela resumindo parte dos efeitos do CFO sobre um sistema OFDM.

Sistema	Estrutura da Matriz de Canal		Interferências
	No domínio do Tempo	No domínio da Frequência	
Sistema não-OFDM	Toeplitz	Sem Estrutura	IBI+ICI
Sistema OFDM, sem CFO	Circulante	Diagonal	Apenas ICI
Sistema OFDM, com CFO	Toeplitz, diagonal corrompida	Sem Estrutura	IBI+ICI+Desvio de Fase

## 4.2.2 Algoritmo ESPRIT para Estimação de CFO para OFDM

**Evolução de Algoritmos para Estimação de CFO** A estimação de CFO em sistemas OFDM desenvolveu-se inicialmente através do uso de portadoras-piloto contendo informações conhecidas [51]. No artigo [52] é apresentado um algoritmo que não utiliza portadoras-piloto. No caso, a estimação é feita com base na análise dos efeitos de CFO simultaneamente nos dados transmitidos e no prefixo cíclico. O estimador utilizado é do tipo ML, cuja implementação apresenta uma complexidade computacional relativamente alta.

Os métodos paramétricos para estimação de CFO exploram redundâncias nas estruturas presentes no sistema OFDM, não necessitam de portadoras-piloto e são alternativas de menor complexidade que o método ML. Em [53] é apresentado um algoritmo que utiliza técnica semelhante ao algoritmo MUSIC com a finalidade de estimar CFO, enquanto no artigo [6] é apresentada outra alternativa, com base no algoritmo ESPRIT.

Ao comparar a evolução dos algoritmos de estimação de CFO com o dos algoritmos de estimação de DoA, apresentado na Seção 2.3.1, observa-se que a flexibilidade na escolha da geometria das antenas no caso da estimação de DoA permitiu o desenvolvimento de muitas alternativas diferentes para estimação paramétrica. Tal alternativa não existe para estimação de CFO em um ambiente OFDM. No caso da

estimação de CFO, o uso de portadoras piloto ou de um prefixo cíclico predominou nos primeiros algoritmos pela existência de ambos em sistemas OFDM, independentemente da presença de um sistema para compensação de CFO. De maneira similar ao caso de estimação de DoA, o uso de métodos paramétricos para estimação de CFO possibilitou o desenvolvimento de métodos com menor complexidade, mais adequados a uma implementação em tempo real.

**Ambiente:** Considere os vetores auxiliares  $\tilde{\mathbf{s}}(k)$  e  $\mathbf{r}(k)$ , ambos com comprimento  $P$ , definidos da seguinte forma

$$\tilde{\mathbf{s}}(k) = \mathbf{H}_{D,PS}(k)e^{j(k-1)\theta(N+L)}. \quad (4.10)$$

$$\mathbf{r}(k) = e^{-j\theta(N-1)}\mathbf{Z}\tilde{\mathbf{s}}^*(k), \quad (4.11)$$

onde a matriz  $\mathbf{Z}$  é definida por

$$\mathbf{Z} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & e^{j2\pi(N-1)/M} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{j2\pi(P-1)(N-1)/M} \end{bmatrix}. \quad (4.12)$$

Seja ainda  $y_i(k)$  o elemento de  $\mathbf{y}(k)$  referente à  $i$ -ésima portadora. Formam-se  $(N - M)$  vetores  $\mathbf{y}_F^{(i)}(k)$  tais que

$$\mathbf{y}_F^{(i)}(k) = \begin{bmatrix} y_i(k) \\ y_{i+1}(k) \\ \vdots \\ y_{i+M}(k) \end{bmatrix}, \quad (4.13)$$

onde  $1 \leq i \leq (N - M)$  e  $P \leq M \leq N - 1$ .

Formam-se ainda outros  $(N - M)$  vetores  $\mathbf{y}_B^{(i)}(k)$  tais que

$$\mathbf{y}_B^{(i)}(k) = \begin{bmatrix} y_{N-i}^*(k) \\ y_{N-i-1}^*(k) \\ \vdots \\ y_{N-i-M}^*(k) \end{bmatrix}. \quad (4.14)$$

O efeito correspondente ao CFO é conjuntamente modelado pelas seguintes matrizes

$$\mathbf{E}_{M+1} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & e^{j\theta} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{jM\theta} \end{bmatrix}, \quad (4.15)$$

$$\mathbf{\Phi} = \begin{bmatrix} e^{j\theta} & 0 & \dots & 0 \\ 0 & e^{j\theta+2\pi/M} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{j\theta+2\pi} \end{bmatrix}, \quad (4.16)$$

$$= e^{j\theta} \begin{bmatrix} 1 & 0 & \dots & 0 & & & \\ 0 & e^{j2\pi/M} & \dots & 0 & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{j2\pi} & & & & \end{bmatrix}, \quad (4.17)$$

onde  $\mathbf{\Phi}$  representa uma redundância presente na modelagem do sistema, e que representa a matriz de parâmetros a ser estimada pelo método ESPRIT.

Como mostrado na Subseção 2.3.1, os métodos de estimação SSI são divididos em 4 etapas, sendo a última etapa associada a uma função SSI4, dependente do ambiente. A função SSI4 para o ambiente DoA foi apresentada na equação (2.54). Para estimação de CFO, a função SSI4 é definida por

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}) = \frac{1}{j} \ln \frac{\text{tr}(\hat{\mathbf{\Phi}})}{\sum_{k=0}^{P-1} e^{jk\omega}}, \quad (4.18)$$

onde  $\text{tr}(\cdot)$  retorna o traço de seu argumento. No caso da estimação de CFO, o vetor de parâmetros estimados  $\theta$  reduz-se a um escalar.

Os vetores  $\mathbf{y}_F^{(i)}(k)$  podem então ser representados por

$$\mathbf{y}_F^{(i)}(k) = \mathbf{E}_{M+1} \mathbf{W}_{M+1}(\mathbf{\Phi})^i \tilde{\mathbf{s}}(k), \quad (4.19)$$

$$\mathbf{y}_B^{(i)}(k) = \mathbf{E}_{M+1} \mathbf{W}_{M+1}(\mathbf{\Phi})^i \mathbf{r}(k), \quad (4.20)$$

onde  $\mathbf{W}_{M+1}$  é uma matriz de dimensões  $(M+1) \times (M+1)$  contendo as  $M+1$  primeiras linhas e colunas de  $\mathbf{W}$ .

Considere então a matriz de covariância  $\mathbf{R}_{YE}$  de dimensões  $P \times P$ , definida por

$$\mathbf{R}_{YE} = \frac{1}{K(N-M)} \sum_{k=1}^K \sum_{i=1}^{N-M} \left[ \mathbf{y}_F^{(i)}(k) (\mathbf{y}_F^{(i)}(k))^H + \mathbf{y}_B^{(i)}(k) (\mathbf{y}_B^{(i)}(k))^H \right], \quad (4.21)$$

onde  $K$  denota o número total de blocos OFDM transmitidos. Definindo-se  $\mathbf{A} = \mathbf{E}_{M+1}\mathbf{W}_{M+1}$ , a equação (4.21) pode ser reescrita como

$$\mathbf{R}_{YE} = \mathbf{A} \left[ \frac{1}{K(N-M)} \sum_{k=1}^K \sum_{i=1}^{N-M} (\tilde{\mathbf{s}}(k)\tilde{\mathbf{s}}^H(k) + \mathbf{r}(k)\mathbf{r}^H(k)) \right] \mathbf{A}^H = \mathbf{A}\mathbf{R}_{\tilde{\mathbf{s}}}\mathbf{A}^H, \quad (4.22)$$

onde  $\mathbf{R}_{\tilde{\mathbf{s}}}$  representa o termo entre colchetes na equação (4.22), sendo portanto uma covariância do tipo FB dos dados de entrada. Além disso,  $\tilde{\mathbf{s}}$  representa uma estrutura *forward* enquanto  $\mathbf{r}(k)$  representa uma estrutura *backward*, mantendo a característica FB da covariância.

Até o momento supõe-se o ambiente sem ruído aditivo. Incorporando-se AWGN ao modelo, tem-se que

$$\mathbf{R}_{YE} = \mathbf{A}\mathbf{R}_{\tilde{\mathbf{s}}}\mathbf{A}^H + \sigma_N^2\mathbf{I}. \quad (4.23)$$

Forma-se então a matriz  $\mathbf{R}_0$ , de dimensões  $M \times M$ , contendo as  $M$  primeiras linhas e as  $M$  primeiras colunas de  $\mathbf{R}_{YE}$ .

São realizadas então as duas primeiras etapas da estimação SSI conforme descrito na Subseção 2.3.1, equações (2.30) e (2.31),

$$\mathbf{Y}_E^{(i)} = \left[ (\mathbf{y}_F^{(i)}(0) + \mathbf{y}_B^{(i)}(0)) \quad (\mathbf{y}_F^{(i)}(1) + \mathbf{y}_B^{(i)}(1)) \quad \dots \quad (\mathbf{y}_F^{(i)}(K-1) + \mathbf{y}_B^{(i)}(K-1)) \right] \quad (4.24)$$

$$\mathbf{Y}_E = \sum_i \mathbf{Y}_E^{(i)}, \quad (4.25)$$

$$\mathbf{R}_0 = \mathbf{J}_X \mathbf{R}_{YE} \mathbf{J}_X^T, \quad (4.26)$$

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_0, \text{SSI1}(\mathbf{Y}_E)). \quad (4.27)$$

Pode-se mostrar que existe uma transformação linear  $\mathbf{T}$ , de posto completo que satisfaz a seguinte equação [6]

$$\mathbf{A} = \mathbf{U}_s \mathbf{T}. \quad (4.28)$$

Define-se então a matriz  $\mathbf{J}_X$  de dimensões  $M \times (M+1)$ , que seleciona  $M$  primeiras linhas de uma matriz quando  $\mathbf{J}_X$  é multiplicada pela esquerda da matriz, ou seja,

$$\mathbf{J}_X = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}. \quad (4.29)$$

Define-se também a matriz  $\mathbf{J}_Y$  de dimensões  $M \times (M+1)$ , que seleciona as  $M$  últimas linhas de uma matriz quando  $\mathbf{J}_Y$  é multiplicada pela esquerda dessa matriz,

ou seja,

$$\mathbf{J}_Y = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}. \quad (4.30)$$

Como o efeito de CFO é modelado como cumulativo ao longo das subportadoras gerando uma redundância na modelagem do sistema, a seguinte relação é verificada,

$$\mathbf{J}_X \mathbf{A} \Phi = \mathbf{J}_Y \mathbf{A}. \quad (4.31)$$

Usando-se a equação (4.28), chega-se a

$$\mathbf{J}_X \mathbf{U}_s \mathbf{T} \Phi = \mathbf{J}_Y \mathbf{U}_s \mathbf{T}, \quad (4.32)$$

ou seja, como  $\mathbf{T}$  tem posto completo,

$$\mathbf{J}_X \mathbf{U}_s \Psi = \mathbf{J}_Y \mathbf{U}_s, \quad (4.33)$$

onde  $\Psi$  é definida como

$$\Psi = \mathbf{T} \Phi \mathbf{T}^{-1}. \quad (4.34)$$

Calcula-se  $\Psi$  a partir de uma estimação TLS definida pela equação (2.38), isto é,

$$[\hat{\Psi}] = \text{TLS}(\mathbf{J}_X \mathbf{U}_s, \mathbf{J}_Y \mathbf{U}_s). \quad (4.35)$$

Como  $\Psi$  e  $\Phi$  apresentam uma relação equivalente às transformações de similaridade [27], ambos possuem os mesmos autovalores, ou seja, elementos da diagonal de  $\Phi$  são os autovalores de  $\Psi$ . Desta forma,  $\Phi$  pode ser obtida através de uma EVD sobre  $\Psi$ .

Por fim, é realizada a etapa 4 de estimação SSI, conforme mostrado na equação (4.18), ou seja,

$$[\hat{\theta}] = \text{SSI4}(\hat{\Phi}). \quad (4.36)$$

De uma maneira alternativa, a estimação do efeito de CFO pode ser obtida através do primeiro termo do  $\Phi$  estimado, isto é,

$$\hat{\theta} = \frac{1}{j} \ln[\Phi_{1,1}]. \quad (4.37)$$

A equação (4.18) realiza uma estimação mais precisa, por utilizar todos os termos de  $\Phi$  na estimação e não apenas o primeiro termo.



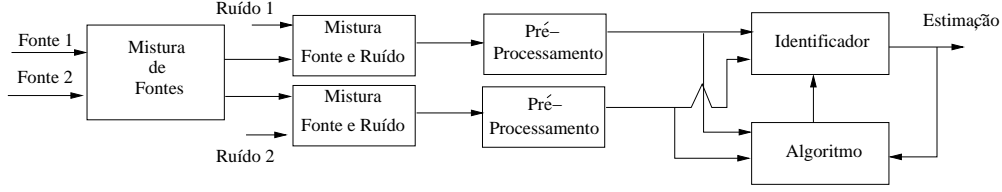


Figura 4.2: Ambiente de identificação que engloba, como casos particulares, os ambientes de equalização cega, de estimação de DoA e de estimação de CFO em OFDM.

### 4.3 Estendendo o TXK para Estimação de CFO

Os algoritmos propostos para estimação foram inicialmente aplicados a um cenário de estimação de DoA. A seguir, o algoritmo CB-DoA foi adaptado para ser aplicado para estimação de CFO em um sistema OFDM. Tal aplicação teve inspiração na versão do algoritmo ESPRIT para estimação de CFO, apresentada na Seção 4.2. Antes de apresentar o algoritmo CB-CFO, será mostrada uma nova extensão do algoritmo TXK para funcionar em um ambiente OFDM.

**Novo Ambiente:** Na Subseção 3.2, o algoritmo TXK, anteriormente apresentado na subseção 2.2.2, tem sua formulação estendida para o ambiente de estimação de DoA. O objetivo desta subseção é fazer a extensão do algoritmo apresentado na Subseção 3.2 para englobar o ambiente de estimação de CFO.

Na Figura 4.2 é mostrado o novo ambiente para o qual o TXK é estendido. Ao contrário dos ambientes anteriores, não há apenas ruído aditivo. Agora há uma interferência multiplicativa gerada pelo CFO. Outra diferença em relação aos ambientes anteriores é que agora há um pré-processamento dos dados antes do algoritmo de estimação, pela necessidade de geração das estruturas  $\mathbf{y}_F^i$  e  $\mathbf{y}_B^i$ , introduzidas nas equações (4.19) e (4.20), respectivamente. A partir de  $\mathbf{y}_F^i$  e  $\mathbf{y}_B^i$ , é então realizada uma estimação do tipo FB, que apresenta uma maior robustez a situações em que a estimação é feita em regiões do espectro com menor resolução [32].

Assim como na Subseção 3.2, considera-se mais uma vez que sinais vetoriais  $\mathbf{s}_1$  e  $\mathbf{s}_2$  são gerados no transmissor. No caso do OFDM, esses vetores representam sinais gerados no domínio da frequência correspondentes a um mesmo usuário. Ambos têm comprimento  $(P - 1)$ , sendo que, para o OFDM, os últimos  $(P - 2)$  componentes

de  $\mathbf{s}_1$  são iguais aos  $(P - 2)$  primeiros componentes de  $\mathbf{s}_2$ . No caso de equalização,  $\mathbf{s}_1$  e  $\mathbf{s}_2$  são gerados pelo mesmo processo estocástico enquanto na estimação de DoA representam diferentes fontes.

As matrizes de mistura  $\mathbf{M}_1$  e  $\mathbf{M}_2$  são definidas a partir das equações (4.10) e (4.19), para o caso OFDM,

$$\mathbf{M}_1 = \mathbf{J}_X \mathbf{E}_{M+1} \mathbf{W}_{M+1} (\Phi)^i \mathbf{H}_{D,P} \mathbf{J}_X^H, \quad (4.38)$$

$$\mathbf{M}_2 = \mathbf{J}_Y \mathbf{E}_{M+1} \mathbf{W}_{M+1} (\Phi)^i \mathbf{H}_{D,P} \mathbf{J}_X^H. \quad (4.39)$$

Diferentemente da extensão anterior do TXK, a interferência não ocorre somente com ruído aditivo, pois há também uma parcela multiplicativa. Os dados recebidos  $\mathbf{x}_1$  e  $\mathbf{x}_2$  obedecem agora as seguintes expressões

$$\mathbf{x}_1 = \mathbf{M}_1 \mathbf{s}_1 e^{j\theta+2\pi/M} + \mathbf{n}_1, \quad (4.40)$$

$$\mathbf{x}_2 = \mathbf{M}_2 \mathbf{s}_2 e^{j\theta+2\pi/M} + \mathbf{n}_2, \quad (4.41)$$

onde  $\mathbf{n}_1 = \mathbf{n}_2$  no caso de estimação de CFO, ou seja, a mesma realização de ruído branco aditivo interfere em  $\mathbf{x}_1$  e  $\mathbf{x}_2$ .

Outra novidade em relação ao algoritmo TXK estendido mostrado na Subseção 3.2 é um pré-processamento de dados no receptor. Tal pré-processamento no contexto de estimação de CFO corresponde à geração de estruturas FBs, mostradas nas equações (4.19) e (4.20), e adaptadas às estruturas desta subseção, conforme mostrado a seguir

$$\mathbf{y}_{F,1} = \mathbf{x}_1, \quad (4.42)$$

$$\mathbf{y}_{B,1} = \mathbf{M}_1 \mathbf{J}_X^T \Lambda^{-1} \mathbf{J}_X \mathbf{s}_1^* e^{j\theta+2\pi/M}, \quad (4.43)$$

$$\mathbf{y}_{F,2} = \mathbf{x}_2, \quad (4.44)$$

$$\mathbf{y}_{B,2} = \mathbf{M}_2 \mathbf{J}_X^T \Lambda^{-1} \mathbf{J}_Y \mathbf{s}_2^* e^{j\theta+2\pi/M}. \quad (4.45)$$

A partir dos vetores  $\mathbf{y}_{F,1}$ ,  $\mathbf{y}_{F,2}$ ,  $\mathbf{y}_{B,1}$  e  $\mathbf{y}_{B,2}$  são obtidos os vetores  $\mathbf{y}_1$  e  $\mathbf{y}_2$  utilizados na estimação do CFO,

$$\mathbf{y}_1 = \mathbf{y}_{F,1} + \mathbf{y}_{B,1}, \quad (4.46)$$

$$\mathbf{y}_2 = \mathbf{y}_{F,2} + \mathbf{y}_{B,2}. \quad (4.47)$$

**Estimação usando nova extensão do TXK:** Considere agora a nova estrutura de covariância  $\mathbf{R}_{y1} = E[\mathbf{y}_1 \mathbf{y}_1^H]$  e a estrutura  $\mathbf{R}_{n1}$  como matriz de autocovariância do ruído  $\mathbf{n}_1$ . Consideram-se as duas primeiras etapas de estimação SSI, como definidos na Subseção 2.3.1,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}, \quad (4.48)$$

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_{y1}, \mathbf{R}_{n1}, \text{SSI1}(\mathbf{Z})), \quad (4.49)$$

Considerando-se as estruturas de covariância  $\mathbf{R}_{y12} = E[\mathbf{y}_1 \mathbf{y}_2^H]$  e  $\mathbf{\Sigma}_{12} = E[\mathbf{n}_1 \mathbf{n}_2^H]$ , tem-se que

$$\mathbf{\Psi} = \mathbf{\Lambda}^{-1} \mathbf{U}_s^H [\mathbf{R}_{y12} - \mathbf{\Sigma}_{12}] \mathbf{U}_s \mathbf{\Lambda}_s = \mathbf{\Lambda}^{-1} \mathbf{U}_s^H \mathbf{M}_1 \mathbf{R}_{s12} \mathbf{\Phi} \mathbf{M}_1^H \mathbf{U}_s \mathbf{\Lambda}_s, \quad (4.50)$$

onde o significado de  $\mathbf{\Phi}$  depende do contexto utilizado para estimação. É feita a operação TB como definida na Subseção 2.2.2,

$$[\mathbf{\Psi}] = \text{TB}(\mathbf{U}_s, \mathbf{\Lambda}_s, \mathbf{R}_{y12} - \mathbf{\Sigma}_{12}). \quad (4.51)$$

Assim como na subseção 3.2,  $\mathbf{\Phi}$  pode ser estimada a partir de uma EVD sobre  $\mathbf{\Psi}$ .

Então é realizada a quarta etapa de estimação SSI, da mesma maneira que a equação (4.37),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (4.52)$$

## 4.4 Covariance-Based CFO Estimation (CB-CFO)

Nesta seção foram utilizadas a modelagem e as estruturas apresentadas na Seção 4.2 para o sistema OFDM. A Tabela 4.2 mostra algumas equivalências entre estruturas apresentadas na Seção 4.3 com relação a estruturas a serem apresentadas nesta seção.

Primeiramente são realizadas as duas primeiras etapas do processo de estimação SSI, de acordo com o desenvolvimento visto na Subseção 2.3.1, equações (2.30) e (2.31)

$$\mathbf{Y}_E = \left[ (\mathbf{y}_F(0) + \mathbf{y}_B(0)) \quad (\mathbf{y}_F(1) + \mathbf{y}_B(1)) \quad \dots \quad (\mathbf{y}_F(K-1) + \mathbf{y}_B(K-1)) \right] \quad (4.53)$$

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_{Y_E}, \text{SSI1}(\mathbf{Y}_E)), \quad (4.54)$$

Tabela 4.2: Equivalências entre TXK estendido e o ambiente de estimação de CFO.

CFO	TXK Estendido
$\mathbf{J}_X \mathbf{Y}_E$	$\mathbf{y}_1$
$\mathbf{J}_Y \mathbf{Y}_E$	$\mathbf{y}_2$
$\mathbf{J}_X \mathbf{R}_{YE} \mathbf{J}_X^T$	$\mathbf{R}_{y1}$
$\mathbf{J}_1 \mathbf{A} \mathbf{J}_1^T$	$\mathbf{M}_1$

onde  $\mathbf{R}_{YE} = \mathbf{Y}_E \mathbf{Y}_E^H$  representa a covariância dos dados FB recebidos.

Pode-se mostrar que existe uma matriz  $\mathbf{V}$ , com posto completo, que satisfaz [4]

$$\mathbf{J}_X \mathbf{A} = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{V}. \quad (4.55)$$

Forma-se a matriz  $\mathbf{R}_{1,0}$  definida por

$$\mathbf{R}_{1,0} = \mathbf{J}_Y \mathbf{R}_{YE} \mathbf{J}_X^T \quad (4.56)$$

Pela definição (4.22),

$$\mathbf{R}_{1,0} = \mathbf{J}_Y \mathbf{A} \mathbf{R}_{\tilde{s}} \mathbf{A}^H \mathbf{J}_X^T \quad (4.57)$$

Percebe-se também uma redundância  $\mathbf{\Phi}$  na estrutura da matriz de CFO de modo que

$$\mathbf{J}_X \mathbf{A} \mathbf{\Phi} = \mathbf{J}_Y \mathbf{A}. \quad (4.58)$$

A partir das equações (4.57) e (4.58), tem-se

$$\mathbf{R}_{1,0} = \mathbf{J}_X \mathbf{A} \mathbf{\Phi} \mathbf{R}_{\tilde{s}} \mathbf{A}^H \mathbf{J}_X^T. \quad (4.59)$$

Usando a equação (4.55), pode-se escrever que

$$\mathbf{R}_{1,0} = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{V} \mathbf{\Phi} \mathbf{R}_{\tilde{s}} \mathbf{V}^H \mathbf{\Lambda}_s^H \mathbf{U}_s^H, \quad (4.60)$$

que pode ser reescrita como

$$\mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{R}_{1,0} \mathbf{U}_s \mathbf{\Lambda}_s = \mathbf{V} \mathbf{\Phi} \mathbf{R}_{\tilde{s}} \mathbf{V}^H. \quad (4.61)$$

É realizada uma operação TB como definida na Subseção 2.2.2,

$$[\mathbf{V} \mathbf{\Phi} \mathbf{R}_{\tilde{s}} \mathbf{V}^H] = \text{TB}(\mathbf{U}_s, \mathbf{\Lambda}_s, \mathbf{R}_{1,0}). \quad (4.62)$$

Como os elementos de  $\mathbf{R}_s$  são reais e a matriz é diagonal,  $\hat{\Phi}$  pode ser estimado através de uma EVD sobre  $\mathbf{R}_{1,0}$ , verificando-se a fase dos autovalores encontrados. Por fim, é realizada a quarta etapa de estimação SSI de acordo com a equação (4.18),

$$[\hat{\theta}] = \text{SSI4}(\hat{\Phi}). \quad (4.63)$$

Também pode ser utilizada a forma alternativa para a função SSI4 representada pela equação (4.37).

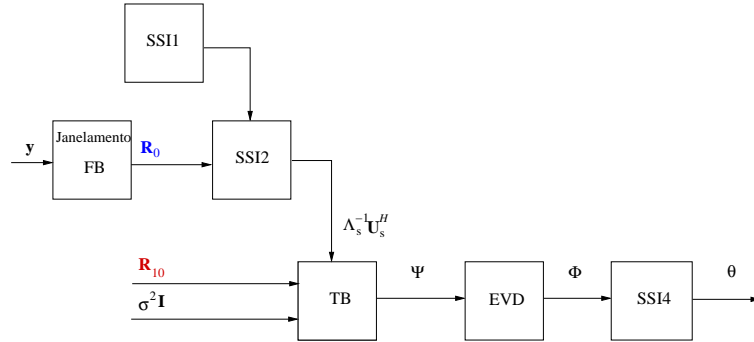


Figura 4.3: Diagrama de blocos representando o processo de estimação do algoritmo CB-CFO.

Este algoritmo para estimação de CFO é aqui denominado *Covariance-Based Carrier Frequency Offset Estimation* (CB-CFO). A Figura 4.3 representa um resumo de operações do algoritmo CB-CFO.

#### 4.4.1 Comparações entre CB-CFO e TLS-ESPRIT

**Complexidade Computacional:** Nesta subseção, o método de estimação paramétrica para CFO descrito na Seção 4.3 será comparado com a versão do ESPRIT adotada para estimação de CFO, descrita em [6]. Uma descrição resumida dos dois algoritmos é encontrada na Tabela 4.3.

Verifica-se que o algoritmo TLS-ESPRIT requer:

- 3 EVDs (1 para matriz hermitiana de dimensões  $2P \times 2P$ , 1 para uma matriz hermitiana de dimensões  $(M+1) \times (M+1)$  e 1 para uma matriz não-hermitiana de dimensões  $P \times P$ );
- 1 inversão de uma matriz  $P \times P$ ;

Tabela 4.3: Comparação dos algoritmos TLS-ESPRIT [6] e CB-CFO.

TLS-ESPRIT	CB-CFO
$[\mathbf{U}_s, \mathbf{\Lambda}_s] = \text{EVD}(\hat{\mathbf{R}})$	$\mathbf{R}_0 = \mathbf{J}_X \hat{\mathbf{R}} \mathbf{J}_X^T$
$\mathbf{E}_0 = \mathbf{J}_X \mathbf{U}_s$	$\mathbf{R}_{1,0} = \mathbf{J}_Y \hat{\mathbf{R}} \mathbf{J}_X^T$
$\mathbf{E}_1 = \mathbf{J}_Y \mathbf{U}_s$	$[\mathbf{U}_s, \mathbf{\Lambda}_s] = \text{EVD}(\mathbf{R}_0)$
$\mathbf{E}_a = \begin{bmatrix} \mathbf{E}_0^H \\ \mathbf{E}_1^H \end{bmatrix} \begin{bmatrix} \mathbf{E}_0 & \mathbf{E}_1 \end{bmatrix}$	$\mathbf{F} = \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H$
$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{E}_a)$	$\mathbf{\Psi} = \mathbf{F} \mathbf{R}_{1,0} \mathbf{F}^H$
$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}$	
$\mathbf{\Psi} = -\mathbf{E}_{12} \mathbf{E}_{22}^{-1}$	
$[\mathbf{T}, \mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$	$[\mathbf{V}, \mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$

- 5 multiplicações de matrizes (4 entre uma matriz de dimensões  $P \times N$  e uma matriz de dimensões  $M \times P$  e 1 entre duas matrizes de dimensões  $P \times P$ ).

Já o algoritmo CB-CFO requer:

- 2 EVDs (1 para uma matriz hermitiana de dimensões  $M \times M$  e 1 para uma matriz hermitiana de dimensões  $P \times P$ );
- 1 inversão de uma matriz diagonal de dimensões  $P \times P$ ;
- 3 multiplicações de matrizes (1 entre uma matriz de dimensões  $P \times P$  e uma matriz de dimensões  $P \times M$ , 1 entre uma matriz de dimensões  $P \times M$  e uma matriz de dimensões  $M \times M$  e 1 entre uma matriz de dimensões  $P \times M$  e uma matriz de dimensões  $M \times P$ ).

Apesar de a implementação ser dependente da plataforma utilizada, pode-se verificar que o algoritmo CB-CFO requer menos operações que o algoritmo TLS-ESPRIT e a diminuição bastante significativa no número de operações ocorre exatamente para as operações mais custosas computacionalmente, como pode ser verificado na Tabela 4.4.

Tabela 4.4: Comparação para as operações requeridas por TLS-ESPRIT e CB-CFO.

Operação	Complexid. [27]	ESPRIT	CB-CFO
EVD Matrizes Não-Hermitianas	$\mathcal{O}(25n^3)$	1	–
EVD Matrizes Hermitianas	$\mathcal{O}(n^2)$	2	2
Inversão	$\mathcal{O}(2n^3/3)$	1	–
Inversão Matr. Diagonais	$\mathcal{O}(n)$	–	1
Multiplicação	$\mathcal{O}(n^3)$	5	3

**Desempenho:** A fim de se verificar a influência da redução de complexidade computacional sobre a taxa de erro de recepção, foram realizadas simulações em ambiente Matlab para ambos os algoritmos.

Os parâmetros utilizados na simulação são listados a seguir:

- Parâmetros derivados de especificações do LTE:
  - Número de portadoras ( $N$ ): 512;
  - Número de portadoras não-virtuais ( $P$ ): 310.
- Tamanho dos dados utilizados ( $M$ ): 511;
- 50 execuções no *ensemble*;
- Canais retirados de especificações do GSM [5]:
  - Rural (R):
 

<b>Atraso (em amostras)</b>	0	2	5	8	16	23	50
<b>Potência (dB)</b>	-3	0	-2	-3	-6	-8	-10
  - Urbano (U):
 

<b>Atraso</b>	0	1	3	5	8	11	13	17	23	31	32	51
<b>(em amostras)</b>												
<b>Potência (dB)</b>	-4	-3	0	-2,6	-3	-5	-7	-5	-6,5	-8,5	-11	-10
- Comprimento  $L = 55$  para o canal urbano e  $L = 11$  para o canal rural;
- Métrica utilizada [6]: *Normalized mean-square error* (NMSE)

$$\text{NMSE} = \frac{1}{Q} \sum_{i=1}^Q \left( \frac{\hat{\theta} - \theta}{2\pi/N} \right)^2, \quad (4.64)$$

onde  $Q$  representa o número de execuções no *ensemble*;

- 200 blocos OFDM transmitidos.

Os resultados das simulações estão representados na Figura 4.4. Nota-se que o desempenho de ambos algoritmos é equivalente para toda a faixa de SNRs medida em ambos os canais, porém, como visto, o algoritmo CB-CFO representa uma complexidade reduzida em relação ao ESPRIT.

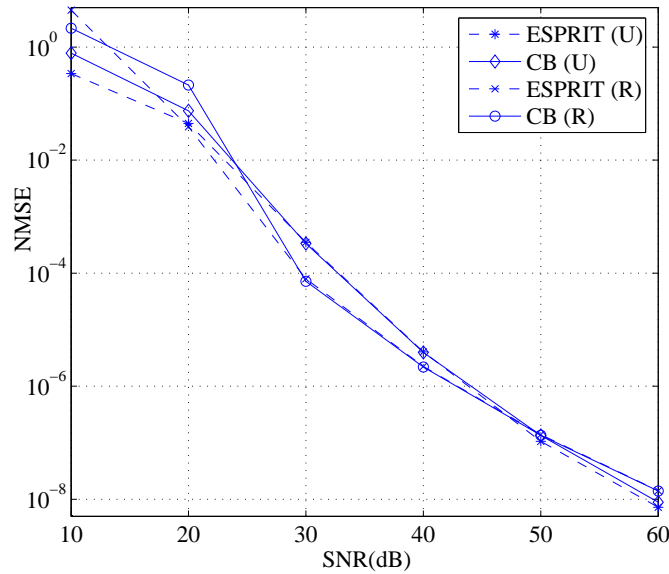


Figura 4.4: NMSE para ESPRIT e CB-CFO.

## 4.5 *Unitary* ESPRIT para Estimação de CFO

Na Subseção 4.2.2 foi apresentada uma versão do ESPRIT para estimação de CFO em ambiente OFDM. O objetivo desta seção é apresentar uma nova versão proposta para o algoritmo ESPRIT nesse mesmo tipo de ambiente. Neste algoritmo realizam-se apenas operações reais sobre os dados recebidos, de maneira semelhante ao *Unitary* ESPRIT utilizado para estimação de DoA, mostrado na Subseção 3.5. O algoritmo *Unitary* ESPRIT impõe uma série de restrições extras sobre a geometria das antenas de recepção. O sistema OFDM oferece espaçamento uniforme entre as portadoras. Tal característica possibilita um processo de estimação equivalentes



ao de uma ULA para a estimação de DoA. Com isso, as restrições CH já estão automaticamente obedecidas pelas características do ambiente utilizado.

**Estruturas utilizadas:** O sistema é modelado pelas equações (4.10) e (4.11), repetidas a seguir por comodidade,

$$\tilde{\mathbf{s}}(k) = \mathbf{H}_{D,PS}(k)e^{j(k-1)\theta(N+L)}. \quad (4.65)$$

$$\mathbf{r}(k) = e^{-j\theta(N-1)}\mathbf{Z}\tilde{\mathbf{s}}^*(k). \quad (4.66)$$

São utilizadas as estruturas auxiliares  $\mathbf{y}_F^{(i)}(k)$  e  $\mathbf{y}_B^{(i)}(k)$ , conforme definidas nas equações (4.13) e (4.14). Numa modelagem inicialmente ignorando o AWGN, o sistema com CFO é definido pela equações (4.19) e (4.20), repetidas a seguir por comodidade,

$$\mathbf{y}_F^{(i)}(k) = \mathbf{E}_{M+1}\mathbf{W}_{M+1}(\Phi)^i\tilde{\mathbf{s}}(k), \quad (4.67)$$

$$\mathbf{y}_B^{(i)}(k) = \mathbf{E}_{M+1}\mathbf{W}_{M+1}(\Phi)^i\mathbf{r}(k), \quad (4.68)$$

Considere agora a estrutura FB  $\mathbf{Y}_E^i$  para todos os  $K$  blocos transmitidos,

$$\mathbf{Y}_E^i = \left[ (\mathbf{y}_F^i(0) + \mathbf{y}_B^i(0)) \quad (\mathbf{y}_F^i(1) + \mathbf{y}_B^i(1)) \quad \dots \quad (\mathbf{y}_F^i(K-1) + \mathbf{y}_B^i(K-1)) \right], \quad (4.69)$$

onde  $0 \leq i \leq (N - M - 1)$ . Define-se uma outra estrutura FB, que soma todos os efeitos de CFO ao longo do bloco,

$$\mathbf{Y}_E = \sum_{i=0}^{N-M-1} \mathbf{Y}_E^i. \quad (4.70)$$

Em comparação com o algoritmo *Unitary* ESPRIT para estimação de DoA, visto na subseção 2.3.5, a estrutura  $\mathbf{Y}_E$  exerce papel equivalente ao que a matriz de dados  $\mathbf{Z}$  exerce para o *Unitary* ESPRIT.

**Transformações CH:** Considere mais uma vez as restrições geométricas CH impostas na Seção 3.5. Para o caso da estimação de CFO em cenário OFDM, a matriz  $\mathbf{E}_{M+1}\mathbf{W}_{M+1}$  desempenha papel equivalente ao do *array manifold* para estimação de DoA. A matriz  $\mathbf{E}_{M+1}$  é diagonal com elementos de um vetor que possui uma estrutura do tipo Vandermonde, enquanto  $\mathbf{W}_{M+1}$ , matriz de IDFT, tem uma estrutura Vandermonde. Então a resultante  $\mathbf{E}_{M+1}\mathbf{W}_{M+1}$  também tem uma estrutura

Vandermonde, o que caracteriza um *array manifold* ULA, que atende as restrições CH para um *array manifold*, de acordo com o apresentado na equação (2.99).

Considere uma matriz  $\mathbf{Q}_a$  de rotação, com dimensões  $a \times a$ , definida pela equação (2.100), repetida a seguir por conveniência,

$$\mathbf{Q}_a = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_{a/2} & j\mathbf{I}_{a/2} \\ \mathbf{\Pi}_{a/2} & -j\mathbf{\Pi}_{a/2} \end{bmatrix}, \quad (4.71)$$

onde  $\mathbf{\Pi}_i$  é uma matriz de dimensões  $i \times i$  com 1 em sua antidiagonal e zero nos demais elementos e  $\mathbf{I}_i$  é uma identidade de dimensões  $i \times i$ . Define-se mais uma vez a transformação  $\mathcal{E}(\cdot)$ , como já mostrada na equação (2.104), repetida a seguir

$$\mathcal{E}(\mathbf{M}) = \begin{bmatrix} \mathbf{M} & \mathbf{\Pi}_a \mathbf{M}^* \mathbf{\Pi}_a \end{bmatrix}, \quad (4.72)$$

onde  $(\cdot)^*$  representa uma operação que retorna o complexo conjugado de cada elemento da matriz. Define-se também a transformação  $\mathcal{T}(\cdot)$  com uma expressão equivalente à equação (2.105),

$$\mathcal{T}(\mathbf{M}) = \mathbf{Q}_a \mathcal{E}(\mathbf{M}) \mathbf{Q}_a^H. \quad (4.73)$$

Considere duas novas matrizes de seleção, geradas da mesma maneira que as equações (3.49) e (3.50)

$$\mathbf{K}_X = \mathbf{Q}_M^H (\mathbf{J}_X + \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{M+1}) \mathbf{Q}_{M+1}, \quad (4.74)$$

$$\mathbf{K}_Y = j \mathbf{Q}_M^H (\mathbf{J}_X - \mathbf{\Pi}_N \mathbf{J}_X \mathbf{\Pi}_{M+1}) \mathbf{Q}_{M+1}. \quad (4.75)$$

**Estimação:** Considere  $\mathbf{R}_{\mathcal{T}}$  como a matriz de covariância de  $\mathcal{T}(\mathbf{Y}_E)$ , ou seja,

$$\mathbf{R}_{\mathcal{T}} = \mathcal{T}(\mathbf{Y}_E) \mathcal{T}^H(\mathbf{Y}_E) \quad (4.76)$$

que possui apenas elementos reais. São realizadas as duas primeiras etapas do processo de estimação SSI, ou seja,

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2b}(\mathbf{R}_{\mathcal{T}}, \text{SSI1}(\mathbf{Y}_E)), \quad (4.77)$$

A partir do subespaço dos sinais, são extraídas duas submatrizes

$$\mathbf{U}_X = \mathbf{K}_X \mathbf{U}_s \quad (4.78)$$

$$\mathbf{U}_Y = \mathbf{K}_Y \mathbf{U}_s \quad (4.79)$$

De maneira equivalente à Subseção 2.3.5, pode ser mostrado que esses subespaços obedecem uma equação de invariância,

$$\mathbf{U}_Y = \mathbf{U}_X \mathbf{\Psi}. \quad (4.80)$$

É encontrada uma solução TLS para a equação (4.80) a partir da equação (2.38), ou seja,

$$[\hat{\mathbf{\Psi}}] = \text{TLS}(\mathbf{U}_X, \mathbf{U}_Y). \quad (4.81)$$

De maneira equivalente ao desenvolvimento presente na Subseção 2.3.5, pode ser mostrado que  $\mathbf{\Psi}$  se relaciona com  $\mathbf{\Phi}$  a partir de

$$\mathbf{\Psi} = \mathbf{T} \mathbf{\Phi} \mathbf{T}^{-1}, \quad (4.82)$$

isto é,  $\mathbf{\Phi}$  pode ser obtido a partir de uma EVD sobre  $\mathbf{\Psi}_{\text{TLS}}$ .

Finalmente, a última etapa de estimação SSI para CFO é obtida a partir da equação (4.18),

$$\hat{\theta} = \text{SSI4}(\hat{\mathbf{\Phi}}). \quad (4.83)$$

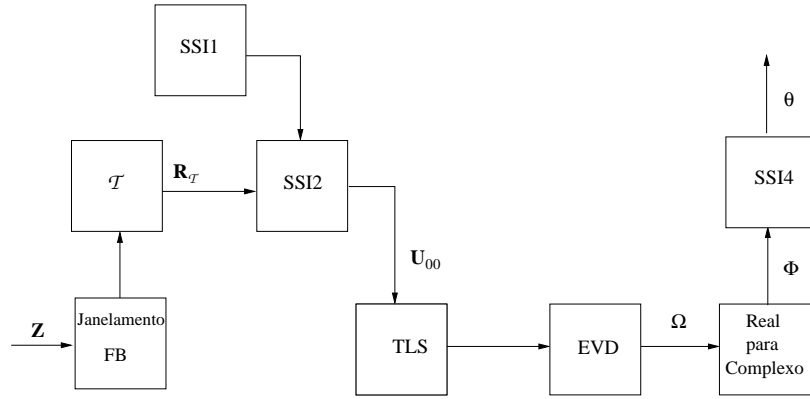


Figura 4.5: Diagrama de blocos representando o processo de estimação do algoritmo *Unitary* ESPRIT.

A Figura 4.5 representa o processo de estimação do algoritmo *Unitary* ESPRIT através de um diagrama de blocos.

### 4.5.1 Comparações entre ESPRIT e *Unitary* ESPRIT para Estimação de CFO

**Complexidade Computacional** Nesta seção será feita uma análise de complexidade do *Unitary* ESPRIT com um tipo de comparação um pouco diferente dos algoritmos anteriores. Tanto os algoritmos da família *covariance-based* (CB) quanto o DD-DoA tinham como proposta uma diminuição de complexidade computacional com relação às suas contrapartidas ESPRIT. A proposta do *Unitary* ESPRIT para estimação de CFO é principalmente apresentar uma alternativa para computação de operações apenas reais, como foi discutido na Seção 3.7. São mostradas algumas comparações com o ESPRIT para estimação de CFO, para se situar computacionalmente a complexidade do *Unitary* ESPRIT. A Tabela 4.5 apresenta um resumo da comparação do *Unitary* ESPRIT com o TLS-ESPRIT, ambos para estimação de CFO.

Verifica-se que o algoritmo TLS-ESPRIT requer:

- 3 EVDs (1 para matriz hermitiana de dimensões  $2P \times 2P$ , 1 para uma matriz hermitiana de dimensões  $(M+1) \times (M+1)$  e 1 para uma matriz não-hermitiana de dimensões  $P \times P$ );
- 1 inversão de uma matriz  $P \times P$ ;
- 5 multiplicações de matrizes (4 entre matrizes de dimensões  $P \times N$  e matrizes de dimensões  $M \times P$  e 1 entre duas matrizes de dimensões  $P \times P$ ).

Todas as operações do TLS-ESPRIT são operações sobre dados complexos.

O algoritmo *Unitary* ESPRIT requer:

- 3 EVDs (1 para matriz hermitiana de dimensões  $2P \times 2P$ , 1 para uma matriz hermitiana de dimensões  $2(M+1) \times 2(M+1)$  e 1 para uma matriz não-hermitiana de dimensões  $P \times P$ );
- 1 inversão de uma matriz  $P \times P$ ;
- 8 multiplicações de matrizes (1 entre uma matriz de dimensões  $2(M+1) \times (M+1)$  e uma matriz de dimensões  $(M+1) \times 2(M+1)$ , 4 entre matrizes de dimensões  $P \times (M+1)$  por matrizes de dimensões  $(M+1) \times P$ , 1 entre duas matrizes

Tabela 4.5: Comparação dos algoritmos TLS-ESPRIT [6] e *Unitary* ESPRIT para estimação de CFO.

<b>TLS-ESPRIT</b>	<b><i>Unitary</i> ESPRIT</b>
$[\mathbf{U}_s, \mathbf{\Lambda}_s] = \text{EVD}(\hat{\mathbf{R}})$	$\mathbf{Y}_E = \sum(\mathbf{y}_B + \mathbf{y}_F)$
$\mathbf{E}_0 = \mathbf{J}_X \mathbf{U}_s$	$\mathbf{\Xi} = \mathbf{\Pi}_M \mathbf{J}_X \mathbf{\Pi}_{M+1}$
$\mathbf{E}_1 = \mathbf{J}_Y \mathbf{U}_s$	$\mathbf{K}_X = \mathbf{Q}_{M+1}^H (\mathbf{J}_X + \mathbf{\Xi}) \mathbf{Q}_{2(M+1)}$
$\mathbf{E}_a = \begin{bmatrix} \mathbf{E}_0^H \\ \mathbf{E}_1^H \end{bmatrix} \begin{bmatrix} \mathbf{E}_0 & \mathbf{E}_1 \end{bmatrix}$	$\mathbf{K}_Y = \mathbf{Q}_{M+1}^H (\mathbf{J}_X - \mathbf{\Xi}) \mathbf{Q}_{2(M+1)}$
$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{E}_a)$	$\mathbf{R}_T = \mathcal{T}(\mathbf{Y}_E) \mathcal{T}^H(\mathbf{Y}_E)$
$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}$	$[\mathbf{U}_s, \mathbf{\Lambda}_s] = \text{EVD}(\hat{\mathbf{R}}_T)$
$\mathbf{\Psi} = -\mathbf{E}_{12} \mathbf{E}_{22}^{-1}$	$\mathbf{U}_0 = \mathbf{K}_X \mathbf{U}_s$
$[\mathbf{T}, \mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$	$\mathbf{U}_1 = \mathbf{K}_Y \mathbf{U}_s$
	$\mathbf{E}_a = \begin{bmatrix} \mathbf{U}_0^H \\ \mathbf{U}_1^H \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 & \mathbf{U}_1 \end{bmatrix}$
	$[\mathbf{E}, \mathbf{\Lambda}] = \text{EVD}(\mathbf{E}_a)$
	$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}$
	$\mathbf{\Psi} = -\mathbf{E}_{12} \mathbf{E}_{22}^{-1}$
	$[\mathbf{T}, \mathbf{\Phi}] = \text{EVD}(\mathbf{\Psi})$

de dimensões  $P \times P$  e 2 entre uma matriz de dimensões  $(M + 1) \times 2(M + 1)$  e uma matriz de dimensões  $2(M + 1) \times P$ .

Dentre essas operações, apenas a multiplicação entre uma matriz de dimensões  $2(M + 1) \times (M + 1)$  e uma de dimensões  $(M + 1) \times 2(M + 1)$  é feita sobre elementos complexos. As demais operações são feitas sobre estruturas reais.

Da mesma maneira que os algoritmos anteriores, a Tabela 4.6 apresenta um resumo das operações feitas pelo algoritmo *Unitary* ESPRIT no contexto de estimação de CFO, em comparação com o ESPRIT. Para o *Unitary* ESPRIT é separada a única operação de multiplicação envolvendo estruturas complexas. Verifica-se que

as operações de inversão e multiplicação são as que o algoritmo *Unitary* ESPRIT apresenta maiores ganhos de complexidade computacional em comparação com o ESPRIT.

Tabela 4.6: Comparação entre o número de operações dos algoritmos ESPRIT e *Unitary* ESPRIT para estimação de CFO.

Operação	ESPRIT		Unitary ESPRIT	
	Complex. [27]	#	Complex.	#
EVD Matrizes Não-Hermitianas	$\mathcal{O}(25n^3)$	1	$\mathcal{O}(25n^3)$	1
EVD Matrizes Hermitianas	$\mathcal{O}(2n^2)$	2	$\mathcal{O}(n^2)$	2
Inversão Completa	$\mathcal{O}(8n^3/3)$	1	$\mathcal{O}(2n^3/3)$	1
Multiplicação	$\mathcal{O}(4n^3)$	5	$\mathcal{O}(n^3)$	7
			$\mathcal{O}(4n^3)$	1

**Desempenho:** Algumas simulações foram realizadas para se verificar o desempenho do algoritmo *Unitary* ESPRIT para estimação de CFO. Foram utilizados parâmetros semelhantes aos listados na comparação entre CB-CFO e TLS-ESPRIT, apresentada na Subseção 4.4.1.

A métrica utilizada foi o NMSE, definido na equação (4.64), e repetido a seguir por conveniência,

$$\text{NMSE} = \frac{1}{Q} \sum_{i=1}^Q \left( \frac{\hat{\theta} - \theta}{2\pi/N} \right)^2. \quad (4.84)$$

Os parâmetros utilizados na simulação são listados a seguir:

- Parâmetros derivados de especificações do LTE:
  - Número de portadoras ( $N$ ): 512;
  - Número de portadoras não-virtuais ( $P$ ): 310;
- Tamanho dos dados utilizados ( $M$ ): 511;
- 50 execuções no *ensemble*;
- Foram utilizados os mesmo canais da subseção 4.4.1, repetidos a seguir, por conveniência,

– Rural (R):

<b>Atraso (em amostras)</b>	0	2	5	8	16	23	50
<b>Potência (dB)</b>	-3	0	-2	-3	-6	-8	-10

– Urbano (U):

<b>Atraso</b>	0	1	3	5	8	11	13	17	23	31	32	51
<b>(em amostras)</b>												
<b>Potência (dB)</b>	-4	-3	0	-2,6	-3	-5	-7	-5	-6,5	-8,5	-11	-10

- Comprimento  $L = 55$  para o canal urbano e  $L = 11$  para o canal rural;

Os resultados das simulações estão representados na Figura 4.6. Nota-se que o desempenho apresenta um menor MSE em relação ao algoritmo *Unitary* ESPRIT. Como visto, o algoritmo *Unitary* ESPRIT também apresenta, em relação ao ESPRIT, a vantagem de somente utilizar operações com estruturas reais. Como o pro-

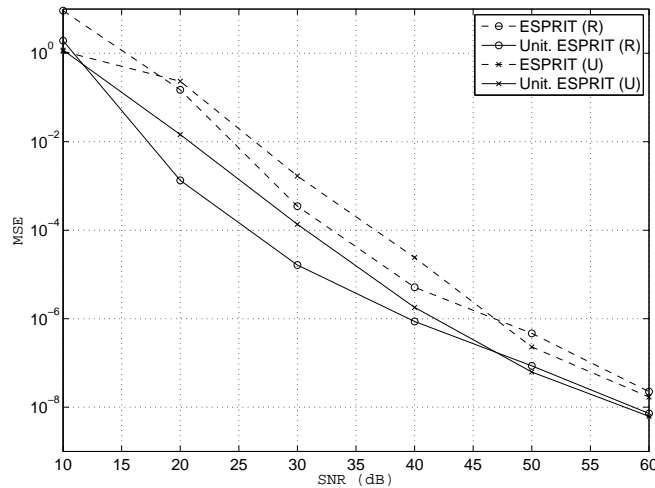


Figura 4.6: NMSE para ESPRIT e *Unitary* ESPRIT para canais rural (R) e urbano (U) das especificações do LTE [5].

cessamento no *Unitary* ESPRIT é feito num domínio distinto do algoritmo ESPRIT, o menor MSE do algoritmo *Unitary* ESPRIT pode ser atribuído a este algoritmo apresentar uma resolução melhor em algumas faixas de ângulos em que o ESPRIT apresente uma resolução ruim.

## 4.6 Discussões do Capítulo

Considerando-se os algoritmos propostos e apresentados nas Seções 3.3 e 4.3, bem como o algoritmo TXK apresentado na Subseção 2.2.2, esta seção tem como objetivo principal apresentar uma discussão das relações entre os algoritmos e um detalhamento de algumas de suas semelhanças e diferenças. O algoritmo TXK estendido descrito nas Subseções 3.2 e 4.3 pode ser considerado como ponto de partida das discussões sobre um *framework* unificado. Começa-se a descrição com um panorama sobre três diferentes contextos de comunicações: equalização, sistemas com múltiplas antenas e OFDM.

Ainda nesta seção são discutidas algumas implementações de algoritmos de estimação de CFO em sistemas comerciais.

**Com Relação aos Algoritmos da Literatura:** A Figura 4.7 apresenta os ambientes OFDM, de estimação de DoA e equalização cega unidos pela ótica de estimação paramétrica. Como já apresentado na seção 2.5, o ambiente de estimação de DoA e equalização cega podem ser modelados como sistemas MIMO. Esse tipo de modelagem pode ser estendida para o caso OFDM se o sinal referente a cada portadora diferente for considerado como uma entrada ou saída diferente no sistema.

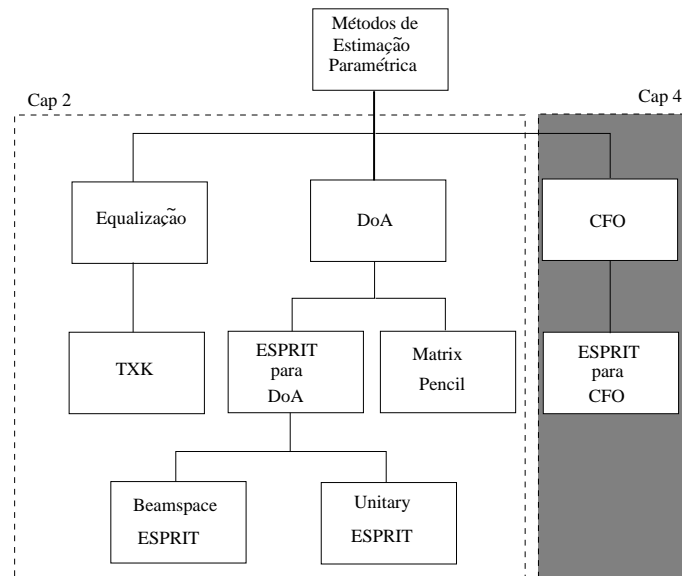


Figura 4.7: Representação dos algoritmos apresentados nos Capítulos 2 e 4, compreendendo os ambientes de equalização cega, estimação de DoA e estimação de CFO para OFDM.



Nos casos de DoA e OFDM, os sinais recebidos correspondem a uma soma de senóides com um ruído de banda larga. A estimação é feita tendo como objetivo um parâmetro da matriz de mistura entre os sinais. Tanto para DoA quanto para OFDM, a estimação leva em conta redundâncias da matriz de mistura. O algoritmo ESPRIT explora tais redundâncias. Por sua vez, no ambiente de equalização, tanto o sinal recebido quanto o ruído aditivo têm banda larga. O sistema de estimação tem como objetivo obter o sinal transmitido, perdendo-se, a princípio, o caráter paramétrico da estimação. No entanto, o algoritmo TXK faz a equalização através de uma identificação de sistema. Essa identificação é obtida através da estimação do vetor de parâmetros  $\mathbf{y}$  na equação (2.25) e (2.26), o que caracteriza a estimação como paramétrica. A descrição do algoritmo TXK na Subseção 2.2.2 teve como objetivo mostrar o *framework* inicial de onde surgiram as ideias para o desenvolvimento dos novos algoritmos.

**Construindo um *Framework* Unificado:** Observa-se que tanto o TXK quanto CB-DoA e CB-CFO apresentam manipulações matemáticas sobre matrizes de covariância do sinal recebido. No caso do TXK, as matrizes utilizadas são  $\mathbf{R}_x(0)$  e  $\mathbf{R}_x(1)$ . Para o CB-DoA, é usada a matriz de autocovariância e a de covariância cruzada, ambas com *lag* zero  $\mathbf{R}_x$  e  $\mathbf{R}_{xy}$ , definidas na Seção 3.3. Por fim, para o CB-CFO, foram utilizadas as estruturas  $\mathbf{J}_X \mathbf{R}_{YE} \mathbf{J}_X^T$  e  $\mathbf{J}_Y \mathbf{R}_{YE} \mathbf{J}_X^T$ , definidos na Seção 4.4.

Aparentemente, as matrizes de covariância usadas no CB-CFO são bem diferentes das matrizes utilizadas para o CB-DoA. No entanto, como apresentado em [32], os conjuntos  $X$  e  $Y$  de antenas não precisam ser disjuntos. Considere, por exemplo, um *array* de antenas de recepção uniformemente espaçado como mostrado na Figura 4.8, onde cada quadrado numerado representa uma antena. Pode-se escolher o *sub-array*  $X$  como o conjunto das antenas com índice ímpar e o *sub-array*  $Y$  como o conjunto das antenas de índice par, satisfazendo a restrição de escolha de *doublets* do CB-DoA. Entretanto, pode-se escolher também os *sub-arrays* da se-

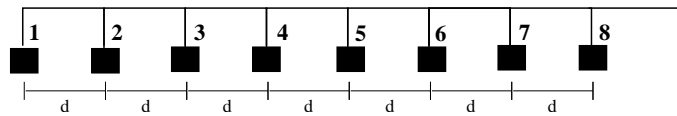


Figura 4.8: *Array* de antenas de recepção.

guinte maneira:  $X = \{1, 2, 3, 4, 5, 6, 7\}$ ,  $Y = \{2, 3, 4, 5, 6, 7, 8\}$ , ou seja, os *doublets* correspondem aos pares de antenas  $(i, i + 1)$  onde  $1 \leq i \leq 7$ , o que também satisfaz as exigências do CB-DoA. Nesse caso, temos que as últimas  $(N - 1)$  linhas de  $\mathbf{R}_x$  equivalem às primeiras  $(N - 1)$  linhas de  $\mathbf{R}_{xy}$ . Ou seja, se adotarmos as estruturas ESPRIT definidas na subseção 2.3.3, têm-se

$$\mathbf{R}_x = \mathbf{J}_X \mathbf{R}_z \mathbf{J}_X^T, \quad (4.85)$$

$$\mathbf{R}_{xy} = \mathbf{J}_Y \mathbf{R}_z \mathbf{J}_X^T. \quad (4.86)$$

Com isso, as matrizes de covariância equivalem às covariâncias no ambiente do algoritmo CB-CFO.

O ruído presente nos algoritmos TXK e CB-CFO é modelado como AWGN, devido à natureza temporal de aquisição do sinal recebido. O fato de serem modelados como brancos é possível devido à natureza completamente aleatória do ruído considerado, enquanto o uso de uma distribuição gaussiana é justificada pelo teorema central do limite (*central limit theorem*) [54]. O ruído aditivo presente no algoritmo CB-DoA não pode ser garantidamente branco, uma vez que há uma amostragem espacial. Por isso é empregada uma matriz genérica de covariância, o que acarreta o uso de uma GEVD. No entanto, por ainda apresentar uma natureza bastante aleatória, supõe-se que o ruído tende a ter amostras com baixa correlação entre si, e sua matriz de covariância aproxima-se de uma diagonal, o que justificaria uma aproximação usando a matriz identidade.

Comparando-se as equações (2.21), (3.21) e (4.57), repetidas aqui, por conveniência,

$$\text{TXK: } \mathbf{R}_x(1) = \mathbf{H}\mathbf{L}\mathbf{H}^H + \sigma_N^2 \mathbf{J}, \quad (4.87)$$

$$\text{CB-DoA: } \mathbf{R}_{xy} = \mathbf{A}\mathbf{\Phi}\mathbf{A}^H + \mathbf{R}_{nxy}, \quad (4.88)$$

$$\text{CB-CFO: } \mathbf{R}_{1,0} = \mathbf{J}_X \mathbf{A}\mathbf{\Phi}\mathbf{R}_{\bar{s}}\mathbf{A}^H \mathbf{J}_X^T, \quad (4.89)$$

percebe-se que, para os algoritmos CB-DoA e CB-CFO, a matriz de covariância está relacionada com a matriz  $\mathbf{\Phi}$  correspondente a cada contexto, ambas diagonais e possuindo posto completo. Já no caso do TXK,  $\mathbf{R}_x(1)$  está relacionada com  $\mathbf{L}$ , que não possui posto completo. É por essa diferença na estrutura das matrizes que é requerido o uso de uma EVD para estimação no caso do CB-DoA e CB-CFO, enquanto o TXK usa uma matriz gerada por uma cadeia de Jordan em seu

processo de estimação, muito mais custosa computacionalmente, por envolver uma SVD e mais uma estimação. Assim, tanto CB-DoA como CB-CFO são algoritmos menos custosos computacionalmente em relação ao TXK, devido às características específicas de seus respectivos problemas.

**Generalized TXK (GTXK)** A partir do que foi apresentado na Subseção 3.2, é possível generalizar o algoritmo TXK, o que pode permitir que se enxergue matematicamente algumas relações existentes entre os algoritmos propostos. Aqui, como em toda a tese, todas as variáveis aleatórias consideradas possuem média zero.

Consideram-se duas estruturas de dados  $\mathbf{x}_1$  e  $\mathbf{x}_2$ , como apresentadas na Subseção 3.2, ou seja, do mesmo modo que nas equações (3.2) e (3.3), desenvolvida para o TXK estendido,

$$\mathbf{x}_1 = \mathbf{M}_1 \mathbf{s}_1 + \mathbf{n}_1, \quad (4.90)$$

$$\mathbf{x}_2 = \mathbf{M}_2 \mathbf{s}_2 + \mathbf{n}_2, \quad (4.91)$$

onde  $\mathbf{s}_1$  e  $\mathbf{s}_2$  são também correlacionadas. Supõe-se que  $\mathbf{M}_2$  tenha as mesmas dimensões que  $\mathbf{M}_1$  e que tenha posto completo por colunas. Além disso, supõe-se que as matrizes  $\mathbf{M}_1$  e  $\mathbf{M}_2$  possam ser relacionadas por uma transformação linear

$$\mathbf{M}_2 = \mathbf{\Phi} \mathbf{M}_1, \quad (4.92)$$

onde  $\mathbf{\Phi}$  é uma matriz quadrada, com suas dimensões iguais ao número de linhas de  $\mathbf{M}_1$  e  $\mathbf{M}_2$ .

Faz-se então um modelo para o sistema baseado em covariâncias,

$$\mathbf{R}_{s1} = E[\mathbf{s}_1 \mathbf{s}_1^H], \quad (4.93)$$

$$\mathbf{R}_{s12} = E[\mathbf{s}_1 \mathbf{s}_2^H], \quad (4.94)$$

$$\sigma^2 \mathbf{\Sigma}_1 = E[\mathbf{n}_1 \mathbf{n}_1^H], \quad (4.95)$$

$$\sigma^2 \mathbf{\Sigma}_{12} = E[\mathbf{n}_1 \mathbf{n}_2^H], \quad (4.96)$$

$$\mathbf{R}_a = E[\mathbf{x}_1 \mathbf{x}_1^H] = \mathbf{M}_1 \mathbf{R}_{s1} \mathbf{M}_1^H + \sigma^2 \mathbf{\Sigma}_1, \quad (4.97)$$

$$\mathbf{R}_b = E[\mathbf{x}_1 \mathbf{x}_2^H] = \mathbf{M}_1 \mathbf{R}_{s1} \mathbf{\Phi}^H \mathbf{M}_1^H + \sigma^2 \mathbf{\Sigma}_{12}, \quad (4.98)$$

onde  $\sigma^2$  representa a potência tanto de  $\mathbf{n}_1$  quanto de  $\mathbf{n}_2$ , ou seja, supõe-se que os ruídos tenham potências iguais.

São realizadas as etapas 1 e 2 de estimação SSI, de acordo com o apresentado na Seção 2.3.1,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}, \quad (4.99)$$

$$[\mathbf{U}_s, \mathbf{\Lambda}_s^2, \sigma_N^2] = \text{SSI2}(\mathbf{R}_{x1}, \mathbf{R}_{n1}, \text{SSI1}(\mathbf{Z})), \quad (4.100)$$

A seguir é feita a operação TB, como definida na Subseção 2.2.2,

$$[\mathbf{R}_c] = \text{TB}(\mathbf{U}_s, \mathbf{\Lambda}_s, \mathbf{R}_b - \sigma^2 \mathbf{\Sigma}_{12}). \quad (4.101)$$

Com isso, pode-se escrever que

$$\mathbf{R}_c = \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{M}_1 \mathbf{R}_{s12} \mathbf{\Phi} \mathbf{M}_1^H \mathbf{U}_s \mathbf{\Lambda}_s. \quad (4.102)$$

É feito um processamento sobre  $\mathbf{R}_c$  para se encontrar  $\mathbf{\Phi}$  que corresponde a uma EVD no caso de estimação de DoA ou de CFO, ou que corresponde a um processamento de uma cadeia de Jordan no caso do TXK. Por fim, é realizada a última etapa de uma estimação SSI, definida na equação (2.54),

$$[\hat{\theta}] = \text{SSI4}(\hat{\mathbf{\Phi}}), \quad (4.103)$$

onde  $\hat{\theta}$  corresponde a um vetor de ângulos no caso de estimação de DoA ou a uma interferência multiplicativa escalar no caso de estimação de CFO, ou dados transmitidos no TXK. O fluxo de operações do algoritmo GTXK está representado na Figura 4.9, a partir das etapas de um algoritmo do tipo SSI, conforme apresentado na Subseção 2.3.1. De certa forma, o algoritmo GTXK pode ser visto como uma generalização também do algoritmo apresentado em [55] para estimação espectral, num contexto de estimação em que a matriz  $\mathbf{\Phi}$  não é necessariamente diagonal, o que engloba o algoritmo TXK.

Considerando-se que no caso do TXK o vetor de parâmetros descoberto seja  $\mathbf{y}$ , os vetores singulares à esquerda associados aos menores valores singulares de  $\mathbf{\Psi}$ , pode-se definir uma operação equivalente à SSI4 a fim de gerar uma estimativa para o canal, representados pelas equações (2.25) e (2.26), reproduzidas a seguir, por conveniência,

$$\mathbf{Q}_2 = \begin{bmatrix} \mathbf{y} & \mathbf{\Psi} \mathbf{y} & \dots & \mathbf{\Psi}^{P-1} \mathbf{y} \end{bmatrix} \quad (4.104)$$

$$\hat{\mathbf{H}} = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{Q}_2 e^{j\phi}. \quad (4.105)$$

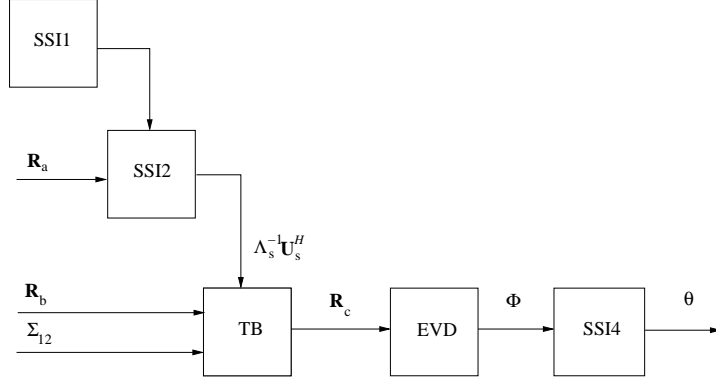


Figura 4.9: Representação do algoritmo GTXK, usando um diagrama de blocos.

Com isso, define-se uma variação na função SSI4:

$$[\hat{\mathbf{H}}] = \text{SSI4b}(\mathbf{y}, \Psi). \quad (4.106)$$

Como a aplicação final do TXK é equalização, usa-se a estimação do canal para se fazer uma equalização ZF, obtendo-se os sinais transmitidos.

**Incorporando uma descrição *beamspace*:** A fim de se agregar a descrição do algoritmo *beamspace* ao algoritmo GTXK, faz-se uma associação de variáveis de maneira direta com o já exposto neste capítulo. Uma pequena modificação é feita na coluna referente ao CB-DoA, considerando-se

$$\mathbf{z}(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{y}(k) \end{bmatrix}, \quad \mathbf{n}_z(k) = \begin{bmatrix} \mathbf{n}_x(k) \\ \mathbf{n}_y(k) \end{bmatrix}, \quad (4.107)$$

ou seja,

$$\mathbf{x}(k) = \mathbf{J}_X \mathbf{z}(k), \quad (4.108)$$

$$\mathbf{y}(k) = \mathbf{J}_Y \mathbf{z}(k), \quad (4.109)$$

$$\mathbf{n}_x(k) = \mathbf{J}_X \mathbf{n}_z(k), \quad (4.110)$$

$$\mathbf{n}_y(k) = \mathbf{J}_Y \mathbf{n}_z(k). \quad (4.111)$$

A diferença no algoritmo para as versões *elementspace* e *beamspace* do CB-DoA tem como origem as diferenças nas respectivas equações de invariância:

$$\textit{Elementspace CB-DoA:} \quad \mathbf{J}_X \mathbf{A} = \mathbf{J}_Y \mathbf{A} \Phi^H, \quad (4.112)$$

$$\textit{Beamspace CB-DoA:} \quad \mathbf{Q} \mathbf{T}_i^H \mathbf{A} = \mathbf{Q} \mathbf{F} \mathbf{T}_i^H \mathbf{A} \Phi^H. \quad (4.113)$$

As matrizes  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  são puramente de seleção, ou seja,  $\mathbf{Z}_1 = \mathbf{J}_X \mathbf{Z}$  e  $\mathbf{Z}_2 = \mathbf{J}_Y \mathbf{Z}$ . O papel exercido por  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  para a versão *elementspace* do CB-DoA é assumido pelas matrizes  $\mathbf{Q}_1 = \mathbf{Q} \mathbf{T}_i^H$  e  $\mathbf{Q}_2 = \mathbf{Q} \mathbf{F} \mathbf{T}_i^H$  na versão *beamspace*. No entanto,  $\mathbf{Q}_1$  e  $\mathbf{Q}_2$  não têm um caráter puramente de seleção. Ao se fazer as operações  $\mathbf{Z}_1 = \mathbf{Q}_1 \mathbf{Z}$  e  $\mathbf{Z}_2 = \mathbf{Q}_2 \mathbf{Z}$ , não é garantido que todas as partes de  $\mathbf{Z}$  estejam preservadas em  $\mathbf{Z}_1$  e  $\mathbf{Z}_2$ . Tendo como foco a estrutura das matrizes, pode-se representar as matrizes de seleção da seguinte maneira

$$\mathbf{J}_X = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}, \quad (4.114)$$

$$\mathbf{J}_Y = \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad (4.115)$$

$$\mathbf{Q}_1 = \begin{bmatrix} \mathbf{Q}_3 & \mathbf{0} \end{bmatrix}, \quad (4.116)$$

$$\mathbf{Q}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{Q}_4 \end{bmatrix}, \quad (4.117)$$

onde  $\mathbf{I}$  representa uma matriz identidade com dimensões apropriadas, enquanto  $\mathbf{Q}_3$  e  $\mathbf{Q}_4$  são matrizes quaisquer de posto completo, de acordo com as especificações definidas na Subseção 3.4. Percebe-se então que  $\mathbf{J}_X$  é um caso específico de  $\mathbf{Q}_1$  e que  $\mathbf{J}_Y$  é um caso específico de  $\mathbf{Q}_2$ . Com isso, a tabela de equivalência de variáveis pode ser novamente reescrita para englobar o caso *BeamSpace* CB-DoA. A Tabela 4.7 mostra essa nova representação, onde  $\mathbf{Q}_1$  e  $\mathbf{Q}_2$  podem ser identificadas com *BeamSpace* CB-DoA ou *ElementSpace* CB-DoA através das equações (4.114) a (4.117)

Tabela 4.7: Nova identificação de entidades do GTXK com TXK, CB-DoA (versões *elementspace* e *beamspace*), e CB-CFO.

GTXK	TXK	CB-DoA	CB-CFO
$\mathbf{x}_1$	$\mathbf{x}(k)$	$\mathbf{Q}_1 \mathbf{z}(k)$	$\mathbf{J}_X \mathbf{Y}_E^{(i)}(k)$
$\mathbf{x}_2$	$\mathbf{x}(k-1)$	$\mathbf{Q}_2 \mathbf{z}(k)$	$\mathbf{J}_Y \mathbf{Y}_E^{(i)}(k)$
$\mathbf{n}_1$	$\mathbf{n}(k)$	$\mathbf{Q}_1 \mathbf{n}_z(k)$	$\mathbf{J}_X \mathbf{n}(k)$
$\mathbf{n}_2$	$\mathbf{n}(k-1)$	$\mathbf{Q}_2 \mathbf{n}_z(k)$	$\mathbf{J}_Y \mathbf{n}(k)$
$\mathbf{s}_1$	$\mathbf{s}(k)$	$\mathbf{J}_X \mathbf{s}(k)$	$\mathbf{J}_X \mathbf{s}(k)$
$\mathbf{s}_2$	$\mathbf{s}(k-1)$	$\mathbf{J}_Y \mathbf{s}(k)$	$\mathbf{J}_Y \mathbf{s}(k)$

## Implementações de Estimação de CFO em Sistemas de Comunicações

Os sistemas OFDM apresentam aplicações em vários sistemas padronizados de co-

municações, como comentado na Seção 1.2. Agora pretende-se destacar algumas implementações referentes especificamente à estimação de CFO em alguns ambientes de comunicações. Na tese [56] são implementados quatro técnicas de estimação de CFO que utilizam sequências-piloto no ambiente padronizado DVB-T (*digital video broadcasting - terrestrial*). Pelas simulações, verificou-se que os algoritmos apresentam uma alta taxa de erro quando ocorrem efeitos Doppler muito severos. O artigo [57] também aborda o mesmo sistema, fazendo um estudo experimental dos efeitos da degradação causada pelo CFO em sistemas DVB-T, utilizando a BER como métrica de comparação.

No artigo [58] é implementado um estimador paramétrico para CFO, com base no algoritmo MUSIC, ou seja, de funcionamento semelhante aos algoritmos a serem apresentados, em uma plataforma de *software-defined radio*, um conceito de flexibilidade de funcionamento que vem ganhando bastante popularidade para implementação de sistemas de comunicações.

Um estudo comparativo de técnicas de estimação de CFO em um ambiente WiMAX é feita no artigo [59]. A tecnologia WiMAX apresenta alternativas para conexão de redes sem fio à Internet que se tornaram bastante utilizadas. A comparação é feita para uma técnica utilizando sinais de treinamento através de portadoras-piloto e outra não-supervisionada, com base em métodos da literatura. É mostrado que a BER de sistemas com compensação de CFO aproxima-se bastante da BER de sistemas livres de CFO. Também é mostrado que sistemas OFDM sem compensação de CFO têm seu desempenho bastante degradado em termos de BER.

## 4.7 Conclusões do Capítulo

Este capítulo aborda o problema de se estimar o *offset* das portadoras de um sistema OFDM. Para esse outro problema de estimação é apresentada uma solução através de um método inspirado no ESPRIT original. Em seguida é mostrado o algoritmo proposto CB-CFO, uma alternativa de menor complexidade que o ESPRIT. Também é apresentado o algoritmo proposto *Unitary* ESPRIT, que utiliza apenas operações reais em seu funcionamento.

Na parte de discussões do capítulo, os algoritmos *Elementspace* CB-DoA, *Be-*

*amspac* CB-DoA, CB-CFO e TXK foram apresentados sob uma ótica unificada na Seção 4.6. A partir dessa ótica pode-se perceber algumas relação matemáticas entre os algoritmos, e entre os ambientes em que são utilizados.

Os algoritmos vistos no Capítulo 3 e neste capítulo, bem como a análise contida no Apêndice A representam as principais contribuições do trabalho de doutorado. O próximo capítulo tem como objetivo apresentar uma visão sobre o trabalho de doutorado, a partir dos algoritmos propostos e das discussões apresentadas.



# Capítulo 5

## Conclusões da Tese

### 5.1 Sobre o Documento

Este documento tem como objetivo apresentar o trabalho desenvolvido durante o doutorado, apresentar algumas reflexões sobre o trabalho realizado, extrair algumas conclusões, bem como fazer algumas propostas de continuação para o trabalho.

No Capítulo 1 são apresentadas as principais motivações e os objetivos do trabalho feito durante o doutorado. O Capítulo 2 apresenta os algoritmos TXK e ESPRIT, nos quais se baseiam os algoritmos propostos. Além disso, são apresentadas as principais famílias de algoritmos de estimação para sistemas com diversidade espacial, e mostradas as principais etapas dos algoritmos baseados em SSI, que é o tipo de técnica em que se podem classificar os algoritmos propostos. No Capítulo 3, são apresentados quatro novos algoritmos para estimação de DoA comparados a diferentes versões do método clássico de estimação paramétrica ESPRIT, tanto em relação à complexidade computacional, quanto a uma métrica MSE. No Capítulo 4 é descrita a extensão dos algoritmos para um ambiente de estimação de CFO para OFDM. Nesse capítulo é apresentada a versão ESPRIT para estimação de CFO, bem como dois algoritmos propostos. Posteriormente, os algoritmos são analisados de maneira a proporcionar um *framework* unificado, e uma generalização do algoritmo TXK é proposta. No Apêndice A será vista uma análise de MSE para os algoritmos CB-DoA e DD-DoA.

## 5.2 Métodos Desenvolvidos

**Contribuições da Tese:** Algoritmos de estimação tiveram papel bastante importante no desenvolvimento de sistemas de comunicações. Em algumas situações, o problema de estimação em comunicações pode ser simplificado para uma estimação do tipo paramétrica, já supondo um modelo subjacente para os sinais, o ambiente e o problema de estimação associado. A abordagem paramétrica permite uma diminuição na complexidade dos algoritmos desenvolvidos. A busca por reduções mais significativas em complexidade computacional levou ao desenvolvimento de algoritmos que exploram a forma de amostragem do ambiente em questão. No caso do ambiente de *array* de antenas, isso se traduz em explorar a geometria do *array* de recepção, como é feito no ESPRIT.

Esta tese gerou quatro algoritmos para estimação de DoA em sistemas com múltiplas antenas (CB-DoA, *BeamSpace* CB-DoA, *Unitary* CB-DoA e DD-DoA), como visto no Capítulo 3 além de dois algoritmos para estimação de CFO em ambiente OFDM (CB-CFO e *Unitary* ESPRIT), apresentado no Capítulo 4. Além disso, conseguiu-se fazer um *template* genérico unificando-se vários desses algoritmos, o que permite enxergar de uma maneira mais matemática as diferenças entre algoritmos e *frameworks*, visto no Capítulo 3. Outra contribuição secundária é uma rápida revisão bibliográfica sobre o uso do ESPRIT em aplicações, presente no Capítulo 2. Além dessas contribuições, foi feito um desenvolvimento matemático para uma expressão de MSE, conforme mostrado no Apêndice A.

A partir dos algoritmos, foram publicados, até o momento, três artigos em congressos internacionais [2] [3] [4] e um em periódico internacional, [1] conseguindo-se a divulgação do trabalho realizado para um público internacional especializado.

Os algoritmos desta tese têm como objetivo levar a busca de redução de complexidade computacional um passo à frente, ao se eliminarem certas operações redundantes presentes no método ESPRIT, através de ideias inspiradas no algoritmo TXK proveniente de um contexto de equalização cega.

Observamos algumas conclusões gerais a respeito dos algoritmos propostos:

- São algoritmos do tipo SSI que alteram a etapa de resolução da equação de invariância, aliviando requisitos computacionais de etapas anteriores;

- Os algoritmos propostos apresentam complexidade computacional menor que a do ESPRIT, que é considerado um algoritmo com complexidade computacional baixa, em relação a seus antecessores tanto paramétricos como não-paramétricos;
- Os algoritmos CB-DoA, *Beam-space* CB-DoA, *Unitary* CB-DoA e CB-CFO apresentam MSE equivalente às suas versões equivalentes para o ESPRIT. No caso do CB-DoA, a equivalência da expressão de MSE é comprovada matematicamente no apêndice A;
- O algoritmo DD-DoA apresenta um compromisso entre complexidade computacional e MSE na comparação com a versão SVD do ESPRIT;
- Os algoritmos propostos partem de premissas a respeito do ambiente e suposições geométricas sobre o *array* de recepção equivalentes ao ESPRIT;
- O algoritmo CB-DoA, tanto em sua versão *elementspace* quanto *beam-space* exige a estimação de matrizes de covariância, não exigidas pelo ESPRIT, que pode operar diretamente sobre os dados;
- O algoritmo CB-DoA apresenta uma redução de complexidade mais pronunciada em relação ao ESPRIT à medida em que se aumenta o número de sensores de recepção. Isso ocorre ao se comparar as versões *elementspace* de cada algoritmo e, também, ao se comparar as versões *beam-space* correspondentes;
- Os algoritmos TXK, *Elementspace* CB-DoA, *Beam-space* CB-DoA e CB-CFO podem ser colocados num *framework* unificado, mostrando a menor complexidade computacional dos algoritmos CB-DoA e CB-CFO devido a características do ambiente no qual estão inseridos.

**Visão pessoal:** O algoritmo ESPRIT apresentou simultaneamente um grande estímulo e um grande desafio para sua modernização. O estímulo consiste na sua popularidade e no grande número de extensões existentes na literatura. De início, o principal obstáculo foi realizar mudanças no ESPRIT que não tivessem sido já realizadas, o que configura um grande desafio devido ao tempo desde o desenvolvimento do ESPRIT, em meados da década de 1980. O desenvolvimento do CB-DoA,

primeiro algoritmo da tese, foi motivado pela busca por uma conexão mais formal entre ESPRIT e TXK e não se encontrarem artigos específicos sobre o tema. Com o tempo, verificou-se uma certa escassez de opções ao ESPRIT que usassem as mesmas restrições quanto à geometria das antenas, o que motivou o desenvolvimento dos demais algoritmos da tese. Na Seção 3.7, foram vistos alguns algoritmos que apresentam algumas características semelhantes ao CB-DoA. Formulações semelhantes trazem a percepção positiva de que várias das ideias contidas nos algoritmos propostos fazem parte do desejo comum de pesquisadores em diminuir a complexidade computacional dos métodos de estimação. Outro ponto positivo é de existirem aspectos diferentes e inovadores nos algoritmos propostos quando comparados aos listados na Seção 3.7, o que estimula o estudo de diversos aspectos computacionais e de desempenho dos algoritmos propostos.

A redução de complexidade proposta para o ESPRIT foi estendida para vários de seus cenários de funcionamento, em particular para os cenários de estimação de DoA e de CFO. Isso leva a acreditar que os novos algoritmos possam ser estendidos a outros cenários e em outras versões, existentes para o ESPRIT, como será indicada na Seção 5.3. O desenvolvimento de uma nova versão do ESPRIT no cenário de estimação de CFO para OFDM mostra que as potencialidades do próprio algoritmo ESPRIT ainda não foram totalmente exploradas, revelando o quanto a área é promissora.

No caso do algoritmo DD-DoA há um compromisso entre redução de complexidade e MSE. Acredita-se que a utilização de uma técnica com uma abordagem bem diferente das existentes deva passar ainda por um período de amadurecimento, o que já ocorreu no caso das técnicas consagradas migradas do ambiente de equalização. Por esse motivo, acredita-se que versões alternativas futuras do DD-DoA possivelmente atinjam uma diminuição da complexidade sem aumento no MSE, como é o caso do CB-DoA, por exemplo.

## 5.3 Propostas de Trabalhos Futuros

A aplicação de métodos paramétricos para estimação em sistemas de comunicações é extensa, como visto no Capítulo 2. A adaptação de nossos métodos de estimação

em novos cenários pode ser útil, em particular em aplicações com requisitos rígidos de desempenho computacional. São apresentadas nesta seção algumas propostas de continuação do trabalho.

Com relação a modificações em algumas partes dos algoritmos propostos:

- As suposições e restrições de nossos algoritmos podem ser trabalhadas de uma maneira diferente. Um exemplo disso é a restrição original de fontes decorrelacionadas do algoritmo CB-DoA, e sua exigência de estimação das matrizes de covariância do ruído;
- Como apresentado na Subseção 2.3.1, os algoritmos propostos fornecem alternativas à etapa de resolução da equação de invariância para algoritmos da família SSI. A fim de diminuir ainda mais a complexidade computacional do sistema, planeja-se estudar otimizações para as demais etapas do processo de estimação, principalmente as que requerem operações com maior complexidade computacional;
- Utilizar o conceito do espaço de Krylov para diminuir a complexidade computacional da cadeia de Jordan no TXK [60];
- Desenvolver um método iterativo de soluções, equivalente ao método SLS (*structured least-squares*) [32], com o objetivo de diminuir ainda mais a complexidade computacional dos algoritmos;
- A redução de complexidade da operação de SVD no DD-DoA pode ser realizada através de algumas técnicas, usadas em outros contextos como o *Power Method* [27] [61].

Com relação ao desenvolvimento de novos algoritmos:

- A modelagem de sistemas com transmissão em banda larga poderia ser bastante útil para novas aplicações em sistemas de telefonia celular. Alguns algoritmos de estimação paramétrica para sistemas em banda larga já existem, como o *Wideband ESPRIT* [12]. No entanto, o desenvolvimento de algoritmos de menor complexidade computacional torna-se ainda mais desejado nestes casos;

- Alguns algoritmos podem ter suas ideias usadas conjuntamente a fim de se obter um ganho ainda maior em complexidade computacional, com isso gerando a variante *Real BeamSpace* CB-DoA, por exemplo;
- Fazer um algoritmo de equalização baseado em DD-DoA, ou seja, operando sem estruturas de covariância;
- Fazer extensões dos algoritmos propostos para que façam *tracking* de posições de fontes ao longo do tempo, como mostrado em [62];
- Desenvolver uma versão do algoritmo CB-CFO que utilize apenas operações reais.

Com relação a contribuições mais teóricas:

- Planeja-se mostrar as relações entre os métodos *BeamSpace* ESPRIT e CB-CFO. Deseja-se também futuramente colocar os algoritmos sob o *framework* dos métodos do tipo MP;
- Desenvolver uma análise de MSE para o algoritmo CB-CFO;
- A prova da proposição apresentada na Subseção 3.7.3.

# Apêndice A

## Análise de MSE

Neste apêndice será mostrada uma análise teórica de MSE para os algoritmos CB-DoA e DD-DoA, de acordo com a abordagem apresentada em [63] para o ESPRIT, com base também no artigo [64]. Algumas outras análises de desempenho do algoritmo ESPRIT, com uma outra abordagem podem ser vistos em [47]. Considera-se o sistema modelado a partir das equações (2.52) e (2.53), reproduzidas aqui por comodidade,

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{n}_x(k), \quad (\text{A.1})$$

$$\mathbf{y}(k) = \mathbf{A}\Phi\mathbf{s}(k) + \mathbf{n}_y(k). \quad (\text{A.2})$$

Além disso, são definidas as estruturas contendo todos os  $K$  *snapshots* transmitidos da mesma maneira que na Seção 3.6, também reproduzidas aqui por comodidade,

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}(0) & \mathbf{s}(1) & \dots & \mathbf{s}(K-1) \end{bmatrix}, \quad (\text{A.3})$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}(0) & \mathbf{x}(1) & \dots & \mathbf{x}(K-1) \end{bmatrix}, \quad (\text{A.4})$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}(0) & \mathbf{y}(1) & \dots & \mathbf{y}(K-1) \end{bmatrix}, \quad (\text{A.5})$$

$$\mathbf{N}_X = \begin{bmatrix} \mathbf{n}_x(0) & \mathbf{n}_x(1) & \dots & \mathbf{n}_x(K-1) \end{bmatrix}, \quad (\text{A.6})$$

$$\mathbf{N}_Y = \begin{bmatrix} \mathbf{n}_y(0) & \mathbf{n}_y(1) & \dots & \mathbf{n}_y(K-1) \end{bmatrix}. \quad (\text{A.7})$$

Neste apêndice, a métrica MSE é definida de uma maneira distinta do restante da tese

$$\text{MSE}(\hat{\theta}_i, \theta_i) = E[|\Delta\theta_i|^2] = E[|\theta_i - \hat{\theta}_i|^2], \quad (\text{A.8})$$

onde  $E[\cdot]$  corresponde ao operador valor estimado e a notação  $\Delta A$  corresponde à diferença entre o valor estimado  $\hat{A}$  e o valor real de uma variável  $A$ . Neste apêndice,

o valor real da variável é considerado determinístico enquanto o valor da estimação é considerado estocástico. O operador  $E[\cdot]$  está associado à operação valor esperado em relação aos valores estimados.

Considere  $\mathbf{z}(k) = [\mathbf{x}(k)^T \mathbf{y}(k)^T]^T$  contendo todo o *snapshot* recebido no instante  $k$ . As matrizes de seleção  $\mathbf{J}_X$  e  $\mathbf{J}_Y$  são definidas da seguinte maneira

$$\mathbf{x}(k) = \mathbf{J}_X \mathbf{z}(k), \quad (\text{A.9})$$

$$\mathbf{y}(k) = \mathbf{J}_Y \mathbf{z}(k). \quad (\text{A.10})$$

**Análise para o DD-DoA** Considere a estrutura auxiliar  $\mathbf{B} = \begin{bmatrix} \mathbf{X} \\ \mathbf{W} \end{bmatrix}$ , a matriz  $\mathbf{V}_B$  é definida contendo os autovetores de  $\mathbf{B}\mathbf{B}^H$  em suas colunas  $\mathbf{v}_i$ . Particiona-se  $\mathbf{V}_B$  tal que

$$\mathbf{V}_X = \mathbf{J}_X \mathbf{V}_B, \quad (\text{A.11})$$

$$\mathbf{V}_W = \mathbf{J}_Y \mathbf{V}_B. \quad (\text{A.12})$$

De acordo com a análise feita em [63], a dependência entre os MSEs correspondentes às estimações dos parâmetros  $\theta$  e  $\phi$  é dada por

$$E[|\Delta\theta_i|^2] = \left( \frac{1}{\omega_i \cos \hat{\theta}_i} \right)^2 \frac{E[|\Delta\phi_i|^2] - \Re[(\hat{\theta}_i)^2 E[(\Delta\phi_i)^2]]}{2}, \quad (\text{A.13})$$

onde  $0 \leq i \leq M - 1$ , e o operador  $\Re[\cdot]$  retorna a parte real de seu argumento.

Em [63] é mostrado que, no caso de apenas duas fontes de transmissão, a aproximação  $E[(\Delta\theta_i)^2] \approx 0$  é válida, o que resulta numa expressão mais simples para  $E[|\Delta\theta_i|^2]$ ,

$$E[|\Delta\theta_i|^2] = \left( \frac{1}{\omega_i \cos \hat{\theta}_i} \right)^2 \frac{E[|\Delta\phi_i|^2]}{2}. \quad (\text{A.14})$$

Então a fim de se ter uma expressão para o MSE no caso de 2 fontes, deve-se obter uma expressão para  $E[|\Delta\phi_i|^2]$  em função das estruturas definidas para o ambiente de estimação de DoA.

Considere  $\Psi$  definido conforme a equação (3.78), repetida a seguir por comodidade,

$$\Psi = \text{SVD}(\mathbf{F}\mathbf{W}) = \mathbf{T}_1((1 + \alpha)\mathbf{I} + \Phi)\mathbf{T}_2^H,$$

onde  $\mathbf{W} = \mathbf{X} + \mathbf{Y}$  e  $\mathbf{F}$  é uma matriz de projeção definida na seção 3.6. Sendo  $\mathbf{r}_i$  os autovetores de  $\Psi$ , ou seja,

$$\Psi \mathbf{r}_i = (1 + \alpha + \phi_i) \mathbf{r}_i. \quad (\text{A.15})$$



Então, numa aproximação de primeira ordem para  $\Delta\phi_i$ ,

$$\Delta\phi_i = \mathbf{r}_i^H (\Delta\mathbf{\Psi}) \mathbf{r}_i. \quad (\text{A.16})$$

A matriz  $\mathbf{\Psi}$  representa a relação de rotação entre subespaços, portanto,

$$\mathbf{\Psi}\mathbf{V}_X = \mathbf{V}_W, \quad (\text{A.17})$$

Assim, a relação entre o erro  $\Delta\mathbf{\Psi}$  com os erros  $\Delta\mathbf{V}_X$  e  $\Delta\mathbf{V}_W$  é expressa por

$$(\mathbf{\Psi} + \Delta\mathbf{\Psi})(\mathbf{V}_X + \Delta\mathbf{V}_X) = \mathbf{V}_W + \Delta\mathbf{V}_W. \quad (\text{A.18})$$

Fazendo-se a aproximação  $(\Delta\mathbf{\Psi})(\Delta\mathbf{V}_X) \approx \mathbf{0}$ , tem-se que

$$\Delta\mathbf{\Psi} = (\Delta\mathbf{V}_W - \mathbf{\Psi}\Delta\mathbf{V}_X)\mathbf{V}_X^\dagger, \quad (\text{A.19})$$

onde a operação  $(\cdot)^\dagger$  denota a pseudo-inversa de Moore-Penrose [27].

Combinando-se as equações (A.15), (A.16) e (A.19),

$$\Delta\phi_i = (\mathbf{r}_i^H \Delta\mathbf{V}_W \mathbf{V}_X^\dagger - (1 + \alpha + \phi_i) \mathbf{r}_i^H \Delta\mathbf{V}_X \mathbf{V}_X^\dagger) \mathbf{r}_i, \quad (\text{A.20})$$

ou ainda,

$$\Delta\phi_i = \mathbf{r}_i^H (\mathbf{J}_Y - (1 + \alpha + \phi_i) \mathbf{J}_X) \Delta\mathbf{V}_B \mathbf{V}_X^\dagger \mathbf{r}_i. \quad (\text{A.21})$$

Com isso, pode-se expressar  $E[|\Delta\phi_i|^2]$  por

$$E[|\Delta\phi_i|^2] = E[(\Delta\phi_i^*)(\Delta\phi_i)] = \mathbf{r}_i^H (\mathbf{V}_X^\dagger)^H E[\Delta\mathbf{V}_B^H \mathbf{C}_i^H \mathbf{r}_i \mathbf{r}_i^H \mathbf{C}_i \Delta\mathbf{V}_B] \mathbf{V}_X^\dagger \mathbf{r}_i. \quad (\text{A.22})$$

onde se define uma nova variável auxiliar  $\mathbf{C}_i = (\mathbf{J}_Y - (1 + \alpha + \phi_i) \mathbf{J}_X)$ . Sendo  $\Delta\mathbf{v}_j$  o erro associado a cada autovetor  $\mathbf{v}_j$  de  $\mathbf{B}\mathbf{B}^H$ , pode-se reescrever a equação (A.22) como

$$E[|\Delta\phi_i|^2] = \mathbf{r}_i^H (\mathbf{V}_X^\dagger)^H \left( \sum_{j=1}^M |\mathbf{r}_i|^2 \mathbf{C}_i^H \mathbf{R}_{\Delta v} \mathbf{C}_i \mathbf{V}_X^\dagger \right) \mathbf{r}_i. \quad (\text{A.23})$$

Em [63] e [64] é mostrada a seguinte relação

$$\mathbf{R}_{\Delta v} = E[\Delta\mathbf{v}_i \Delta\mathbf{v}_j^H] = \frac{\sigma_i}{N} \sum_{k=1, k \neq i} \frac{\sigma_i}{(\sigma_k - \sigma_i)^2} \mathbf{v}_k \mathbf{v}_k^H \delta(i - j), \quad (\text{A.24})$$

onde  $\delta(\cdot)$  representa o impulso de Kronecker. A equação (A.23) pode ser expressa alternativamente por

$$E[|\Delta\phi_i|^2] = \mathbf{r}_i^H (\mathbf{V}_X^\dagger)^H \left( \sum_{j=1}^M |\mathbf{r}_i|^2 \mathbf{C}_i^H \frac{\sigma_i}{N} \sum_{k=1, k \neq i} \frac{\sigma_i}{(\sigma_k - \sigma_i)^2} \mathbf{v}_k \mathbf{v}_k^H \delta(i - j) \mathbf{C}_i \right) \mathbf{V}_X^\dagger \mathbf{r}_i, \quad (\text{A.25})$$

onde  $\sigma_i^2$  representam os autovalores de  $\mathbf{\Psi}$ . Então, após substituir a equação (A.25) na expressão (A.14), chega-se a

$$E[|\Delta\theta_i|^2] = \left(\frac{1}{\omega_i \cos \hat{\theta}_i}\right)^2 \frac{1}{2} \mathbf{r}_i^H (\mathbf{V}_X^H)^+ \left( \sum_{j=1}^M |\mathbf{r}_j|^2 \mathbf{C}_j^H \right. \\ \left. \frac{\sigma_i}{N} \sum_{k=1, k \neq i} \frac{\sigma_i}{(\sigma_k - \sigma_i)^2} \mathbf{v}_k \mathbf{v}_k^H \delta(i-j) \mathbf{C}_i \right) \mathbf{V}_X^+ \mathbf{r}_i, \quad (\text{A.26})$$

que é a aproximação de MSE para  $M = 2$  fontes. Tal equação é próxima da expressão de MSE para o ESPRIT no caso de  $M = 2$  fontes mostrado em [63]. O MSE ligeiramente maior para o DD-DoA ocorre devido à definição diferente do ESPRIT para  $\mathbf{C}_i$ , que, no caso do ESPRIT, é definido como  $\mathbf{C}_i^{\text{ESPRIT}} = \mathbf{J}_X - \phi_i^* \mathbf{J}_X$ .

O algoritmo DD-DoA opera sobre valores singulares, que são valores apenas reais [27]. Por isso, cada  $\phi_i$  tem apenas sua parte real estimada. A parte imaginária de  $\phi_i$  é estimada a partir de sua parte real, aumentando ligeiramente o valor de MSE, devido a erros acumulativos.

**Análise para o CB-DoA** O desenvolvimento para o algoritmo CB-DoA é semelhante ao apresentado em [63] para o ESPRIT. Inicialmente considera-se mais uma vez a modelagem apresentada pela equações (2.52) e (2.53), e apresentadas no início deste apêndice.

Define-se a matriz  $\mathbf{Z} = [\mathbf{X}^T \ \mathbf{Y}^T]^T$  contendo em suas colunas todos os *snapshots* recebidos. Considere  $\mathbf{V}_Z$  como a matriz contendo os autovetores de  $\mathbf{Z}\mathbf{Z}^H$  em suas colunas  $\mathbf{v}_i$ . Particiona-se agora  $\mathbf{V}_Z$  de maneira que

$$\mathbf{V}_X = \mathbf{J}_X \mathbf{V}_Z, \quad (\text{A.27})$$

$$\mathbf{V}_Y = \mathbf{J}_Y \mathbf{V}_Z. \quad (\text{A.28})$$

Do mesmo modo que no caso do DD-DoA, será usada a expressão (A.14) como relação entre  $E[|\Delta\theta_i|^2]$  e  $E[|\Delta\phi_i|^2]$  para o caso de 2 fontes no sistema, ou seja,

$$E[|\Delta\theta_i|^2] = \left(\frac{1}{\omega_i \cos \hat{\theta}_i}\right)^2 \frac{E[|\Delta\phi_i|^2]}{2}.$$

Considera-se a equação (3.23), isto é,

$$\mathbf{\Psi} = \mathbf{T}_1 \mathbf{\Phi}^H \mathbf{T}_1^H,$$

e mais uma vez define-se  $\mathbf{r}_i$  como um autovetor de  $\mathbf{\Psi}$  da mesma maneira que na equação (A.15).

Numa aproximação de primeira ordem para  $\Delta\phi_i$ , equivalente à equação (A.16),

$$\Delta\phi_i = \mathbf{r}_i^H (\mathbf{\Delta\Psi}) \mathbf{r}_i. \quad (\text{A.29})$$

De maneira equivalente à equação (A.18),

$$(\mathbf{\Psi} + \mathbf{\Delta\Psi})(\mathbf{V}_X + \mathbf{\Delta V}_X) = \mathbf{V}_Y + \mathbf{\Delta V}_Y. \quad (\text{A.30})$$

Usa-se mais uma vez a aproximação  $\mathbf{\Delta\Psi}\mathbf{\Delta V}_X \approx \mathbf{0}$ , de modo que

$$\mathbf{\Delta\Psi} = (\mathbf{\Delta V}_Y - \mathbf{\Psi}\mathbf{\Delta V}_X)\mathbf{V}_X^+, \quad (\text{A.31})$$

A partir das equações (A.29), (A.31) e da definição de autovalor,

$$\Delta\phi_i = (\mathbf{r}_i^H \mathbf{\Delta V}_Y \mathbf{V}_X^+ - \phi_i \mathbf{r}_i^H \mathbf{\Delta V}_X \mathbf{V}_X^+) \mathbf{r}_i, \quad (\text{A.32})$$

$$= \mathbf{r}_i^H (\mathbf{J}_X - \phi_i^* \mathbf{J}_X) \mathbf{\Delta V}_X \mathbf{V}_X^+ \mathbf{r}_i. \quad (\text{A.33})$$

Define-se a variável auxiliar  $\mathbf{D}_i = (\mathbf{J}_X - \phi_i^* \mathbf{J}_X)$ . Sendo  $\mathbf{\Delta v}_j$  o erro associado a cada autovetor  $\mathbf{v}_j$  de  $\mathbf{Z}\mathbf{Z}^H$ , então

$$E[|\Delta\phi_i|^2] = \mathbf{r}_i^H (\mathbf{V}_X^+)^H \left( \sum_{j=1}^M |\mathbf{r}_i|^2 \right) \mathbf{D}_i^H \mathbf{R}_{\Delta v} \mathbf{D}_i \mathbf{V}_X^+ \mathbf{r}_i, \quad (\text{A.34})$$

onde  $\mathbf{R}_{\Delta v}$  é dado pela equação (A.24). Então

$$E[|\Delta\phi_i|^2] = \mathbf{r}_i^H (\mathbf{V}_X^+)^H \left( \sum_{j=1}^M |\mathbf{r}_i|^2 \mathbf{D}_i^H \frac{\sigma_k}{N} \sum_{i=1, i \neq k} \frac{\sigma_k}{(\sigma_i - \sigma_k)^2} \mathbf{v}_i \mathbf{v}_i^H \delta(k - j) \mathbf{D}_i \right) (\mathbf{V}_X^H)^+ \mathbf{r}_i, \quad (\text{A.35})$$

onde  $\sigma_i^2$  representam os autovalores de  $\mathbf{\Psi}$ . Substituindo-se em (A.14) chega-se a

$$E[|\Delta\theta_i|^2] = \left( \frac{1}{\omega_i \cos \hat{\theta}_i} \right)^2 \frac{1}{2} \mathbf{r}_i^H (\mathbf{V}_X^H)^+ \left( \sum_{j=1}^M |\mathbf{r}_i|^2 \mathbf{D}_i^H \frac{\sigma_i}{N} \sum_{k=1, k \neq i} \frac{\sigma_i}{(\sigma_k - \sigma_i)^2} \mathbf{v}_k \mathbf{v}_k^H \delta(i - j) \mathbf{D}_i \right) \mathbf{V}_X^+ \mathbf{r}_i, \quad (\text{A.36})$$

que é a expressão final para o MSE do CB-DoA, bastante semelhante à encontrada para o algoritmo ESPRIT em [63].

# Publicações do Autor

- [1] FERREIRA, T. N., NETTO, S. L., DINIZ, P. S. R., “Covariance-Based Direction-of-Arrival Estimation with Real Structures”, *IEEE Signal Processing Letters*, v. 15, pp. 757–760, December 2008.
- [2] FERREIRA, T. N., NETTO, S. L., DINIZ, P. S. R., “Low Complexity Covariance-Based DOA Estimation Algorithm”. In: *Proceedings of the 15th European Signal Processing Conference*, pp. 100–104, Poznan, Poland, September 2007.
- [3] FERREIRA, T. N., NETTO, S. L., DINIZ, P. S. R., “Beamspace Covariance-Based DoA Estimation”. In: *Proceedings of the 9th IEEE Workshop on Signal Processing Advances for Wireless Communications*, pp. 136–140, Recife, Brasil, July 2008.
- [4] FERREIRA, T. N., NETTO, S. L., DINIZ, P. S. R., et al., “Low Complexity Blind Estimation of the Carrier Frequency Offset in Multicarrier Systems”. In: *Proc. of the 33rd IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3045–3048, Las Vegas, April 2008.

# Referências Bibliográficas

- [5] 3GPP GERAN: RADIO TRANSMISSION AND RECEPTION, *3GPP TS 45.005 v7.8.0*, Tech. rep., 3rd Generation Partnership Project, December 2006.
- [6] TURELI, U., LIU, H., ZOLTOWSKI, M. D., “OFDM Blind Carrier Offset Estimation: ESPRIT”, *IEEE Transactions on Communications*, v. 48, n. 9, pp. 1459–1461, September 2000.
- [7] TREES, H. V., *Detection, Estimation and Modulation Theory Part I*. New York, John Wiley and Sons, 1968.
- [8] RAPPAPORT, T. S., *Wireless Communications: Principles and Practice*. 2 ed. Upper Saddle River, Prentice Hall, 2002.
- [9] GODARA, L. C., “Applications of Antenna Arrays to Mobile Communications I. Performance improvement, feasibility and system considerations”, *Proceedings of the IEEE*, v. 85, n. 7, pp. 1031–1060, July 1997.
- [10] DAHLMAN, E., PARKVALL, S., SKÖLD, J., et al., *3G Evolution: HSPA and LTE for Mobile Broadband*. 2 ed. Burlington, Academic Press, 2007.
- [11] PAULRAJ, A., PAPADIAS, C., “Space-Time Processing for Wireless Communications”, *IEEE Signal Processing Magazine*, v. 14, pp. 49–83, November 1997.
- [12] OTTERSTEN, B., KAILATH, T., “Direction-of-arrival estimation for wide-band signals using the ESPRIT algorithm”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 38, n. 2, pp. 317–327, February 1990.
- [13] BALTAR, L. G., *Desvio da Frequência Central em Sistemas Multiportadora*, Master’s Thesis, PEE/COPPE - UFRJ, Rio de Janeiro, Fevereiro 2006.
- [14] ROY, R., KAILATH, T., “ESPRIT - Estimation of Parameters via Rotational Invariance Techniques”, *IEEE Transactions on Acoustics Speech and Signal Processing*, v. 37, n. 7, pp. 984–995, July 1989.

- [15] DINIZ, P. S. R., *Adaptive Filtering: Algorithms and Practical Implementation*. 3 ed. Boston, Springer, 2008.
- [16] TONG, L., XU, G., KAILATH, T., “Blind Identification and Equalization Based on Second-Order Statistics: a Time-Domain Approach”, *IEEE Transactions on Information Theory*, v. 40, n. 2, pp. 340–349, March 1994.
- [17] PROAKIS, J., *Digital Communications*. 3 ed. New York, McGraw-Hill, 2001.
- [18] HUA, Y., SARKAR, T. K., “Matrix Pencil Method for Estimating Parameters of Exponentially Damped/Undamped Sinusoids in Noise”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 38, n. 5, pp. 814–824, May 1990.
- [19] DING, Z., *Blind Equalization and Identification*. New York, Marcel Dekker, 2001.
- [20] CHI, C.-Y., FENG, C.-C., CHEN, C.-H., et al., *Blind Equalization and System Identification*. London, Springer, 2006.
- [21] GARDNER, W., “Exploitation of Spectral Redundancy in Cyclostationary Signals”, *IEEE Signal Processing Magazine*, v. 8, n. 2, pp. 14–36, April 1991.
- [22] GODARA, L. C., “Applications of Antenna Arrays to Mobile Communications II Beam-forming and Direction of Arrival considerations”, *Proceedings of the IEEE*, v. 85, n. 8, pp. 1195–1245, August 1997.
- [23] SCHMIDT, R. O., “Multiple Emitter Location and Signal Parameter Estimation”, *IEEE Trans. on Antennas and Propagation*, v. AP-34, pp. 276–280, March 1986.
- [24] TREES, H. V., *Detection, Estimation and Modulation Theory Part IV, Optimum Array Processing*. New York, John Wiley and Sons, 2002.
- [25] XU, G., SILVERSTEIN, S. D., ROY, R., et al., “Beamspace ESPRIT”, *IEEE Transactions on Signal Processing*, v. 42, n. 2, pp. 349–356, February 1994.
- [26] VAN DER VEEN, A. J., DEPRETTERE, E. F., SWINDLEHURST, A. L., “Subspace-Based Signal Analysis Using Singular Value Decomposition”, *Proceedings of the IEEE*, v. 81, n. 9, pp. 1277–1308, September 1993.
- [27] GOLUB, G., LOAN, C. V., *Matrix Computations*. 3 ed. New York, Johns Hopkins University Press, 1996.

- [28] STOICA, P., NEHORAI, A., “MUSIC, Maximum Likelihood and Crámer-Rao Bound”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 37, n. 5, pp. 720–741, May 1989.
- [29] KUMARESAN, R., TUFTS, D., “Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. ASSP-35, n. 12, pp. 833–840, December 1982.
- [30] RAO, D., HARI, K., “Performance Analysis of Root-MUSIC”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 37, n. 12, pp. 1939–1949, December 1989.
- [31] COLARES, R. F., *Descrição Unificada de Métodos de Estimaco de DoA em Arranjo de Sensores*, Ph.D. Thesis, FEEC/Unicamp, Campinas, Setembro 2000.
- [32] HAARDT, M., *Efficient One-, Two-, and Multidimensional High-Resolution Array-Signal Processing*, Ph.D. Thesis, NWS - Technische Universitaet Muenchen, Muenchen, 1997.
- [33] WAX, M., KAILATH, T., “Detection of signals by information theoretic criteria”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 33, n. 2, pp. 387–392, April 1985.
- [34] YANG, B., “Projection Approximation Subspace Tracking”, *IEEE Transactions on Signal Processing*, v. 43, n. 1, pp. 95–107, January 1995.
- [35] MARKOVSKI, I., VAN HUFFEL, S., “Overview of Total Least-Squares Methods”, *Elsevier Signal Processing*, v. 87, pp. 2283–2302, October 2007.
- [36] DUDGEON, D., JOHNSON, D., *Array Signal Processing: Concepts and Techniques*. Upper Saddle River, Prentice Hall, 1993.
- [37] HAARDT, M., NOSSEK, J. A., “Unitary ESPRIT: How to obtain increased estimation accuracy with reduced computational burden”, *IEEE Transactions on Signal Processing*, v. 43, n. 5, pp. 1232–1242, May 1995.
- [38] ROY, R., PAULRAJ, A., KAILATH, T., “ESPRIT - A Subspace Rotation Approach to Estimation of Parameters of Cisoids in Noise”, *IEEE Trans. on Acoustics, Speech, Signal Processing*, v. ASSP-34, pp. 1340–1342, October 1986.

- [39] SHAN, T.-J., WAX, M., KAILATH, T., “On Spatial Smoothing for Direction-of-Arrival Estimation of Coherent Signals”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. ASSP-33, pp. 806–811, August 1985.
- [40] GUERRA, M., *Utilização de Antenas Adaptativas em Sistemas CDMA*, Master’s Thesis, CETUC/PUC-Rio, Rio de Janeiro, Setembro 2001.
- [41] CAVALCANTI, F. R. P., *Antenas Inteligentes e Processamento Espaço-Temporal para Sistemas de Comunicação Sem Fio*, Ph.D. Thesis, FEEC/Unicamp, Campinas, Julho 1999.
- [42] TSOULOS, G., “Experimental and theoretical capacity analysis of space-division multiple access (SDMA) with adaptive antennas”, *IEE Proceedings on Communications*, v. 146, pp. 307–311, October 1999.
- [43] TSOULOS, G., BEACH, M., MCGEEHAN, J., “Wireless Personal Communications for the 21st Century: European Technological Advances in Adaptive Antennas”, *IEEE Communications Magazine*, v. 35, pp. 102–109, September 1997.
- [44] ALMIDFA, K., TSOULOS, G., NIX, A., “Performance Analysis of ESPRIT, TLS-ESPRIT and Unitary ESPRIT algorithms for DoA Estimation in a WCDMA Mobile System”. In: *Proceedings of the International Conference on 3G Mobile Communication Technologies*, pp. 200–203, March 2000.
- [45] ALMIDFA, K., TSOULOS, G., NIX, A., “Performance Evaluation of Direction-of-Arrival (DoA) estimation algorithms for mobile communication systems”. In: *Proceedings of the Vehicular Technology Conference - Spring*, pp. 1055–1059, May 2000.
- [46] DE LAMARE, R., SAMPAIO NETO, R., “Blind Joint Estimation of Channel and Direction of Arrival using Antenna Arrays in DS-CDMA Systems”. In: *Proceedings of the IEEE Asyomar Conference on Signals, Systems and Computers*, pp. 1005–1009, November 2005.
- [47] OTTERSTEIN, B., VIBERG, M., KAILATH, T., “Performance Analysis of Total Least Squares ESPRIT Algorithm”, *IEEE Transactions on Signal Processing*, v. 39, pp. 1122–1135, May 1991.
- [48] YILMAZER, N., KOH, J., SARKAR, T., “Utilization of a Unitary Transform for Efficient Computation in the Matrix Pencil Method to find the Direction of



- Arrival”, *IEEE Transactions on Antennas and Propagation*, v. 54, n. 1, pp. 175–181, January 2006.
- [49] ZOLTOWSKI, M., “Novel Techniques for Estimation of Array Signal Parameters Based on Matrix Pencils, Subspace Rotations and Total Least-Squares”. In: *Proceedings of the 16th. International Conference on Acoustics, Speech and Signal Processing*, v. 5, pp. 2861–2864, April 1988.
- [50] ZOLTOWSKI, M., STAVRINIDES, D., “Sensor Array Signal Processing via a Procrustes Rotations-based Eigen-analysis of the ESPRIT data Pencil”, *IEEE Transactions on Acoustics, Speech and Signal Processing*, v. 37, n. 6, pp. 832–861, June 1989.
- [51] TURELI, U., KIVANC, D., LIU, H., “Experimental and Analytical Studies on a High-Resolution OFDM Carrier Frequency Offset Estimator”, *IEEE Transactions on Vehicular Technology*, v. 50, n. 2, pp. 629–643, March 2001.
- [52] MOOSE, P., “A Technique for Orthogonal Frequency Division Multiplexing Frequency Offset Correction”, *IEEE Transactions on Communications*, v. 42, n. 10, pp. 2908–2914, October 1994.
- [53] LIU, H., TURELI, U., “A High-Efficiency Carrier Estimator for OFDM Communications”, *IEEE Signal Processing Letters*, v. 2, n. 4, pp. 104–106, April 1998.
- [54] PAPOULIS, A., *Probability, Random Variables and Stochastic Processes*. 3 ed. New York, McGraw-Hill, 1991.
- [55] RAZAVILAR, J., LI, Y., LIU, K. J. R., “Spectral Estimation Based on Structured Low Rank Matrix Pencil”. In: *Proceedings of the 22nd. International Conference on Acoustics, Speech and Signal Processing*, v. 5, pp. 2503–2506, May 1996.
- [56] DIAS, J. T., *Desempenho de Algoritmos para Estimaco de Parmetros de Sincronizao em Sistemas OFDM*, Master’s Thesis, IME, Rio de Janeiro, Abril 2006.
- [57] GALLARDO, A. M., WOODWARD, M. E., RODRIGUEZ-TELLEZ, J., “Performance of DVB-T OFDM based Single Frequency Networks: Effects of Frame Synchronisation, Carrier Frequency Offset and Non-Synchronised Sampling Errors”. In: *Proceedings of the 54th Vehicular Technology Conference - VTC Fall*, pp. 962–966, October 2001.

- [58] TURELI, U., LIU, H., “Software Radio Implementation of Carrier Offset Estimation for OFDM Communications”. In: *Proceedings of the 32nd Asilomar Conference on Signals, Systems and Computers*, pp. 60–64, November 1998.
- [59] GONZÁLEZ-BAYÓN, J., CARRERAS, C., FERNÁNDEZ-HERRERO, A., “Comparative Evaluation of Carrier Frequency Offset Tracking Schemes for WiMAX OFDM Systems”. In: *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology*, pp. 479–484, December 2007.
- [60] IPSEN, I. C. F., MEYER, C. D., “The Idea Behind Krylov Methods”, *American Mathematical Monthly*, v. 105, pp. 889–899, 1997.
- [61] BACKX, F. D., VINHOZA, T. T. V., SAMPAIO-NETO, R., “Power Techniques for Blind Channel Estimation in Zero-Padded OFDM Systems”. In: *Proceedings of the 18th. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–5, Atenas, Grécia, September 2007.
- [62] CASTRO, R. M., DIAS, M. H. C., APOLINÁRIO JR., J. A., “On the Performance of Constrained Adaptive Algorithms for Combined Beamforming and AoA Tracking of a Moving Target”. In: *Proceedings of the International Microwave and Optoelectronics Conference*, pp. 274–278, Brasilia, July 2005.
- [63] RAO, B. D., HARI, K. V. S., “Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise”, *IEEE Trans. on Acoustics, Speech, Signal Processing*, v. 37, pp. 1990–1995, December 1989.
- [64] KAVEH, M., BARABELL, A. J., “The Statistical Performance of the MUSIC and the Minimum-Norm Algorithms in Resolving Plane Waves in Noise”, *IEEE Trans. on Acoustics, Speech, Signal Processing*, v. ASSP-34, pp. 331–341, April 1986.