



Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Programa de Pós-Graduação em Engenharia Elétrica

Tese de Doutorado

Alinhamento de Fase de Portadora para Implementação de
***Beamforming* Distribuído em Redes de Sensores**

Angilberto Muniz Ferreira Sobrinho

Área de Conhecimento: Processamento da Informação

Campina Grande – PB

Fevereiro – 2021



Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Programa de Pós-Graduação em Engenharia Elétrica

Alinhamento de Fase de Portadora para Implementação de *Beamforming* Distribuído em Redes de Sensores

Angilberto Muniz Ferreira Sobrinho

Tese de Doutorado apresentada à Coordenadoria do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande - Campus de Campina Grande como parte dos requisitos necessários para obtenção do grau de Doutor em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento da Informação

José Sérgio da Rocha Neto, D. Sc.

Orientador

Edmar Candeia Gurjão, D. Sc.

Orientador

Campina Grande, Paraíba, Brasil

©Angilberto Muniz Ferreira Sobrinho, Fevereiro de 2021

F383a Ferreira Sobrinho, Angilberto Muniz.
Alinhamento de fase de portadora para implementação de *beamforming* distribuído em redes de sensores / Angilberto Muniz Ferreira Sobrinho. - Campina Grande, 2021.
106 f. : il. Color

Tese (Doutorado em Engenharia Elétrica) - Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, 2021.
"Orientação: Prof. Dr. José Sérgio da Rocha Neto, Prof. Dr. Edmar Candeia Gurjão".
Referências.

1. Redes de Sensores Sem Fio. 2. *Beamforming*. 3. Arranjo Retrodiretivo de Antenas. 4. SDR. 5. GNU - Rádio. 6. *Transponders*. I. Rocha Neto, José Sérgio. II. Gurjão, Edmar Candeia. III. Título.

CDU 621.39:004 (043)

Angilberto Muniz Ferreira Sobrinho

**Alinhamento de Fase de Portadora para Implementação
de *Beamforming* Distribuído em Redes de Sensores**

Tese de Doutorado apresentada à Coordenadoria do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande - Campus de Campina Grande como parte dos requisitos necessários para obtenção do grau de Doutor em Ciências no Domínio da Engenharia Elétrica.

Aprovado em 26 de Fevereiro de 2021.

BANCA EXAMINADORA

Prof. Dr. José Sérgio da Rocha Neto - UFCG
Orientador

Prof. Dr. Edmar Candeia Gurjão - UFCG
Coorientador

Prof. Dr. Angelo Perkusich - UFCG
Membro interno

Prof. Dr. Alexandre Cunha Oliveira - UFCG
Membro interno

Prof. Dr. Charles Casimiro Cavalcante - UFC
Membro externo

Prof. Dr. Vicente Ângelo de Souza Júnior - UFRN
Membro externo

Campina Grande

2021



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
POS-GRADUACAO EM ENGENHARIA ELETRICA
Rua Aprígio Veloso, 882, - Bairro Universitário, Campina Grande/PB, CEP 58429-900

REGISTRO DE PRESENÇA E ASSINATURAS

1. ATA DA DEFESA PARA CONCESSÃO DO GRAU DE DOUTOR EM CIÊNCIAS, NO DOMÍNIO DA ENGENHARIA ELÉTRICA, REALIZADA EM 26 DE FEVEREIRO DE 2021
(Nº 322)

CANDIDATO(A): **ANGILBERTO MUNIZ FERREIRA SOBRINHO**. COMISSÃO EXAMINADORA: ANGELO PERKUSICH, D.Sc., UFCG, Presidente da Comissão, JOSÉ SÉRGIO DA ROCHA NETO, D.Sc., UFCG, EDMAR CANDEIA GURJAO, D.Sc., UFCG, Orientadores, ALEXANDRE CUNHA OLIVEIRA, D.Sc., UFCG, CHARLES CASIMIRO CAVALCANTE, Dr., UFC, VICENTE ÂNGELO DE SOUSA JÚNIOR, Dr., UFRN. TÍTULO DA TESE: Alinhamento de fase de portadora para implementação de beamforming distribuído em redes de sensores. ÁREA DE CONCENTRAÇÃO: Processamento da Informação. HORA DE INÍCIO: **09h00** – LOCAL: **ambiente Virtual, em virtude da suspensão de atividades na UFCG decorrente do coronavírus e e de conformidade com o Art. 8º da PORTARIA PRPG/GPR Nº 003, DE 18 DE MARÇO DE 2020**). Em sessão pública, após exposição de cerca de 45 minutos, o(a) candidato(a) foi arguido(a) oralmente pelos membros da Comissão Examinadora, tendo demonstrado suficiência de conhecimento e capacidade de sistematização, no tema de sua tese, obtendo conceito APROVADO. Face à aprovação, declara o(a) presidente da Comissão, achar-se o examinado, legalmente habilitado(a) a receber o Grau de Doutor em Ciências, no domínio da Engenharia Elétrica, cabendo a Universidade Federal de Campina Grande, como de direito, providenciar a expedição do Diploma, a que o(a) mesmo(a) faz jus. Na forma regulamentar, foi lavrada a presente ata, que é assinada por mim, ÂNGELA DE LOURDES RIBEIRO MATIAS, e os membros da Comissão Examinadora presentes. Campina Grande, 26 de Fevereiro de 2021.

ÂNGELA DE LOURDES RIBEIRO MATIAS
Secretário(a)

ANGELO PERKUSICH, D.Sc., UFCG
Presidente da Comissão e Examinador Interno

JOSÉ SÉRGIO DA ROCHA NETO, D.Sc., UFCG
Orientador

EDMAR CANDEIA GURJÃO, D.Sc., UFCG
Orientador

CHARLES CASIMIRO CAVALCANTE, Dr., UFC

Examinador Externo

VICENTE ÂNGELO DE SOUSA JÚNIOR, Dr., UFRN
Examinador Externo

ALEXANDRE CUNHA OLIVEIRA, D.Sc., UFCG
Examinador Interno

ANGILBERTO MUNIZ FERREIRA SOBRINHO
Candidato

2 - APROVAÇÃO

2.1. Segue a presente Ata de Defesa de Tese de Doutorado do candidato **ANGILBERTO MUNIZ FERREIRA SOBRINHO**, assinada eletronicamente pela Comissão Examinadora acima identificada.

2.2. No caso de examinadores externos que não possuam credenciamento de usuário externo ativo no SEI, para igual assinatura eletrônica, os examinadores internos signatários **certificam** que os examinadores externos acima identificados participaram da defesa da tese e tomaram conhecimento do teor deste documento.



Documento assinado eletronicamente por **ANGELA DE LOURDES RIBEIRO MATIAS, SECRETÁRIO (A)**, em 01/03/2021, às 10:49, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



Documento assinado eletronicamente por **JOSE SERGIO DA ROCHA NETO, VICE-DIRETOR DE CENTRO**, em 01/03/2021, às 11:11, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



Documento assinado eletronicamente por **EDMAR CANDEIA GURJAO, PROFESSOR 3 GRAU**, em 01/03/2021, às 11:27, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



Documento assinado eletronicamente por **ALEXANDRE CUNHA OLIVEIRA, PROFESSOR(A) DO MAGISTERIO SUPERIOR**, em 01/03/2021, às 15:10, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



Documento assinado eletronicamente por **ANGELO PERKUSICH, PROFESSOR(A) DO MAGISTERIO SUPERIOR**, em 01/03/2021, às 15:15, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



Documento assinado eletronicamente por **Angilberto Muniz Ferreira Sobrinho, Usuário Externo**, em 31/03/2021, às 14:58, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.



A autenticidade deste documento pode ser conferida no site <https://sei.ufcg.edu.br/autenticidade>, informando o código verificador **1309653** e o código CRC **00B60143**.

Referência: Processo nº 23096.010000/2021-34

SEI nº 1309653

Caro Professor
Maurício Beltrão de Rossiter Corrêa
Coordenador do PPgEE/UFCG

Eu, **Charles Casimiro Cavalcante, Dr., UFC**, na qualidade de membro da comissão examinadora da defesa de tese intitulada Alinhamento de fase de portadora para implementação de beamforming distribuído em redes de sensores, de **Angilberto Muniz Ferreira Sobrinho**, da qual participei por videoconferência, no dia 26 de fevereiro de 2021, declaro ter recebido cópia da ata, que consta nos autos do processo nº [23096.010000/2021-34](#) como documento nº (1309653) , a qual li e, embora não tenha podido assinar, é para todos os efeitos como se eu tivesse, pois está em perfeita conformidade com o que foi deliberado pela comissão e, portanto, estou de pleno acordo com seus termos.



Charles Casimiro Cavalcante, Dr., UFC

Caro Professor
Maurício Beltrão de Rossiter Corrêa
Coordenador do PPgEE/UFCG

Eu, **Vicente Ângelo de Sousa Júnior, Dr., UFRN**, na qualidade de membro da comissão examinadora da defesa de tese intitulada Alinhamento de fase de portadora para implementação de beamforming distribuído em redes de sensores, de **Angilberto Muniz Ferreira Sobrinho**, da qual participei por videoconferência, no dia 26 de fevereiro de 2021, declaro ter recebido cópia da ata, que consta nos autos do processo nº [23096.010000/2021-34](#) como documento nº (1309653) , a qual li e, embora não tenha podido assinar, é para todos os efeitos como se eu tivesse, pois está em perfeita conformidade com o que foi deliberado pela comissão e, portanto, estou de pleno acordo com seus termos.


Vicente Ângelo de Sousa Júnior, Dr., UFRN

Dedicatória

*A minha mulher Ivana e meus filhos Anderson, Carolina e Nina pela paciência,
compreensão e incentivo ao longo dessa jornada.*

"1o ano: vou desenvolver um conceito que vai mudar a forma como vemos o universo.

2o ano: ... acho que vou restringir o escopo. Um impacto local é suficiente...

3o ano: ... melhor focar apenas neste problema específico mesmo...

4o ano: ... tenho que concluir minha tese..."

(ecg)

"'... a Figura mostra...'

A figura não mostra nada rapaz, quem mostra é você !"

(zs)

"No começo é difícil, mas do meio pro fim, parece que está no começo..."

(igm)

Agradecimentos

Aos meus orientadores, os professores José Sérgio da Rocha Neto e Edmar Candeia Gurgão pela orientação e apoio. Obrigado pelo empenho na disponibilização da infraestrutura necessária para a realização desse trabalho. Obrigado pelos conselhos e comentários técnicos e, sobretudo pela paciência.

À Universidade Federal de Campina Grande, pelo o apoio técnico e estrutural. Ao meu companheiro Tony do laboratório (sala 08), sempre diligente e não medindo esforços para que os instrumentos estivessem disponíveis e operacionais. Ao meu colega Simões, o 'mago' das coisas impossíveis pela construções dos meus circuitos impressos. E, finalmente, ao meu camarada Bruno (2A) pelos momentos de descontração.

À Escola Superior de Tecnologia (EST) da Universidade do Estado do Amazonas (UEA) que me concedeu afastamento para participação neste programa de doutorado.

À COPELE, em especial Angela, Felipe e Pedro, pelos serviços prestados durante o curso deste trabalho acadêmico e pela gentileza e cordialidade.

À CAPES, através da Fundação de Amparo e Pesquisa do Amazonas (FAPEAM) pelo fomento durante o último ano dessa pesquisa.

À minha família que me apoiou incondicionalmente e se sacrificou durante todo o período em que estive envolvido neste trabalho.

Por fim, agradeço a todos que contribuíram de forma direta e indireta para realização deste trabalho.

Resumo

Nessa tese o uso de *beamforming* distribuído em Redes de Sensores Sem Fio é investigado a partir de uma visão macro da tecnologia. Após uma revisão da literatura em que são identificadas duas categorias de abordagens para implementação de *beamforming* distribuído, um método representativo de cada categoria é descrito e um método original para alinhamento de fase das portadoras de transmissão é apresentado. Esse novo método utiliza uma abordagem baseada no conjugado da fase de chegada de um sinal de sincronismo enviado pelo nó interrogador para os nós sensores. Esse método foi implementado e validado experimentalmente em uma plataforma de SDR (Software Defined Radio).

Palavras-chave: *Beamforming*, Redes de Sensores, Arranjo Retrodiretivo de Antenas, SDR, GNU-Radio, *Transponders*.

Abstract

In this thesis, the application of distributed beamforming in WSN is investigated and a macro vision of the technology is presented. After a literature review, two approaches categories for implementation are identified and, in addition to the description of one method representative of each category, an original method for carrier phase alignment based on the phase conjugation of a pilot signal sent from the base station is presented. This method has been implemented and validated experimentally in a SDR (Software Defined Radio) platform.

Keywords: Beamforming, Wireless Sensor Networks, Retrodirective Antenna Array, Software Defined Radio, GNU-Radio, Transponders.

Lista de Siglas

- ADC** - Analog do Digital Converter (Conversor Analógico-Digital)
- AWGN** - Additive White Gaussian Noise (Ruído Branco Gaussiano Aditivo)
- BB** - Base Band (Banda Base)
- BER** - Bit Error Rate (Taxa de Erro de Bit)
- BPSK** - Binary Phase Shift Keying (Chaveamento Binário de Fase)
- CDMA** - Code Division Multiple Access (Acesso Múltiplo por Divisão de Código)
- DAC** - Digital do Analog Converter (Conversor Digital-Analógico)
- FPGA** - Field Programmable Gate Array (Arranjo de Portas Programável em Campo)
- FSM** - Finite State Machine (Máquina de Estados Finitos)
- GPS** - Global Positioning System (Sistema de Posicionamento Global)
- IIR** - Infinite Impulse Response (Resposta Infinita ao Impulso)
- MS/s** - Mega Samples per second (Milhões de Amostras por segundo)
- NCO** - Numerically Controlled Oscillator (Oscilador Controlado Numericamente)
- PDoA** - Phase Difference of Arrival (Diferença de Fase de Chegada)
- PLL** - PLL (Phase Locked Loop)
- RF** - Radio Frequency (Frequência de Rádio)
- RSSF** - Redes de Sensores Sem Fio
- RSSI** - Received Signal Strength Indicator (Indicador de Nível de Sinal Recebido)
- SDR** - Software Defined Radio (Rádio Definido por Software)
- SNR** - Signal to Noise Ratio (Relação Sinal/Ruído)
- TCP/IP** - Transmission Control Protocol / Internet Protocol (Protocolo de Controle de Transmissão)
- TDD** - Time-Division Duplex (Duplexação por Divisão de Tempo)
- uP/uC** - Micro Processor/Controller (Micro Processador/Controlador)
- USRP** - Universal Software Radio Peripheral (Periférico Universal para Rádio definido por Software)
- WSN** - Wireless Sensor Networks (Redes de Sensores sem Fio)

Lista de Tabelas

2.1	Diferenças entre <i>beamforming</i> distribuído e centralizado	14
4.1	Índices de ganho de <i>Beamforming</i> do método RT e do método proposto, IRF	61
4.2	Índices de erro de fase no método (RT) e no método proposto, (IRF)	61
4.3	Comparação entre os métodos One-Bit Feedback (OBF), Time-slotted Round-trip (RT) e o método proposto, Inversão Remota de Fase (IRF)	62

Lista de Figuras

1.1	Representação de uma Rede de Sensores Sem Fio típica.	2
1.2	Diagrama representativo de uma RSSF com colaboração via retransmissão.	2
1.3	RSSF inoperante por falha no último repetidor da rede.	3
1.4	RSSF fora do alcance do coletor	3
1.5	RSSF com colaboração via <i>beamforming</i> distribuído	4
1.6	Representação de um enlace de rádio com multi-percurso. Fonte: reproduzido de [12].	5
1.7	Variações de potência causadas por multi-percurso. Fonte: reproduzido de [12].	5
1.8	Interferências construtivas e destrutivas.	6
1.9	Direcionamento de feixe em Arranjo de Antenas (<i>Phased Array</i>).	7
1.10	Comparação entre a) <i>Beamforming</i> centralizado e b) <i>Beamforming</i> distribuído	8
2.1	Principais métodos de <i>beamforming</i> distribuído.	17
2.2	Esquema simplificado para o método <i>One bit Feedback</i>	22
2.3	Diagrama de rede com 1 Interrogador e N sensores.	25
2.4	Esquema para sincronismo <i>Round Trip</i> . Fonte: Adaptado de [17].	26
2.5	Caminhos de propagação no esquema <i>Round-Trip</i>	27
2.6	Sequência de propagação no método <i>Time-Slotted Round-Trip</i>	28
2.7	Sequência temporal do método <i>Time-Slotted Round-Trip</i> . Fonte: Adaptado de [18].	28
3.1	Diagrama de rede com 1 interrogador e N sensores.	33
3.2	Modelo simplificado do sistema.	34

3.3	Processo de alinhamento de fase pela inversão da fase de chegada do sinal piloto.	35
3.4	Combinação não-coerente do sinal no interrogador devido à inversão, indesejada, da fase inicial e desconhecida do oscilador local.	38
3.5	Representação da inversão remota de fase, considerando $\phi_{Tx_i} = \phi_{Rx_i} = \phi_i$	39
3.6	Estágio de calibração.	41
3.7	Estágio de operação.	42
3.8	Relação temporal: (a) <i>Round-trip</i> , (b) Inversão Remota de Fase.	43
4.1	Diagrama de rede com 1 interrogador e N sensores.	45
4.2	Diferença de fase entre o sinal de <i>beacon</i> enviado pelo interrogador e o sinal recebido por cada sensor.	46
4.3	Simulação do processo de alinhamento de sinal por inversão de fase.	47
4.4	Relação de fase entre os sinais durante recepção combinada no destino.	47
4.5	Resultado da simulação de <i>Beamforming</i> Distribuído via Inversão de Fase	48
4.6	Avaliação da taxa de erro de bits em função da Relação Sinal/Ruído	49
4.7	Diagrama em blocos do arranjo experimental.	50
4.8	Diagrama de blocos dos receptores para o experimento IRF com fase conhecida.	50
4.9	Alinhamento de fase.	51
4.10	Relação de fase entre os sinais durante recepção combinada no destino.	52
4.11	Resultado do método IRF quando a fase é conhecida.	52
4.12	Diagrama de blocos dos receptores para o experimento IRF.	53
4.13	Relação de fase entre os sinais dos sensores no receptor (sem correção de fase).	54
4.14	Recepção combinada sem correção de fase.	54
4.15	Relação de fase entre os sinais dos sensores no receptor (após a correção de fase).	55
4.16	Recepção combinada com correção de fase.	55
4.17	Experimento IRF - Intensidade do sinal recebido pelo interrogador.	56
4.18	Distribuição de ganho e erro de fase (campanha composta de 100 ciclos).	56
4.19	Diagrama de blocos dos receptores para o método RT.	57
4.20	Experimento RT - Intensidade do sinal recebido pelo interrogador.	57

4.21	Distribuição de ganho e erro de fase (campanha com 100 ciclos).	58
4.22	Diagrama em blocos da implementação do experimento <i>One Bit Feedback</i> . . .	58
4.23	Variação de fase e intensidade do sinal no experimento <i>One Bit Feedback</i> . . .	59
4.24	Comparação entre os resultados dos métodos IRF e OBF.	60
4.25	Comparação entre os resultados dos métodos RT e IRF.	61
4.26	Variação de ganho nos métodos IRF e RT.	62
A.1	Estrutura de aplicações desenvolvidas em GNU-Radio.	69
A.2	Diagrama de blocos da plataforma de hardware USRP N2x0. Fonte: reproduzido de [63].	70
A.3	Blocos básicos do arcabouço.	71
A.4	Diagrama de blocos hierárquico da implementação do PLL.	73
A.5	Diagrama de blocos de aplicação ilustrativa (<i>transponder</i>).	75
A.6	Diagrama de blocos do interrogador para o método OBF.	76
A.7	Diagrama de blocos do interrogador para os métodos RT e IRF.	78
A.8	Diagrama de blocos dos transceptores para o método OBF.	79
A.9	Diagrama de blocos dos transceptores para o método RT.	80
A.10	Diagrama de blocos dos transceptores para o experimento IRF com fase conhecida.	81
A.11	Diagrama de blocos dos transceptores para o método IRF.	83
A.12	Diagrama esquemático (<i>grafo</i>) da implementação dos transceptores em GNU-Radio para o método proposto. A parte tracejada é mantida inativa no modo Mestre.	85
A.13	Fluxograma representando a dinâmica do método IRF. O Mestre ignora o estágio de calibração, iniciando diretamente no estágio de operação.	85
B.1	Relação Linear entre deslocamento de fase e frequência.	89
B.2	Diagrama de blocos do transceptor para o método <i>Inversão de Fase baseado em Duplo Tom</i> (IFDT).	91
B.3	Erro de estimativa de fase em função da relação entre f_1 e f_2	92
B.4	Intensidade do sinal recebido no interrogador sem alinhamento de fase. . . .	93
B.5	Detalhe do sinal recebido no interrogador sem alinhamento de fase.	94

B.6	Intensidade do sinal recebido no interrogador com alinhamento de fase. . . .	94
B.7	Detalhe do sinal recebido no interrogador com alinhamento de fase.	95

Sumário

1	Introdução	1
1.1	Redes de Sensores Sem Fio (RSSF)	1
1.2	<i>Beamforming</i>	4
1.3	Motivação	8
1.4	Objetivos	8
1.4.1	Objetivos específicos	9
1.5	Contribuições	9
1.5.1	Principais contribuições - <i>beamforming</i> distribuído	9
1.5.2	Outras contribuições - plataforma SDR	9
1.5.3	Publicações	10
1.6	Estrutura do Documento	10
2	Revisão da Literatura	12
2.1	Definição do Problema	12
2.2	<i>Beamforming</i> distribuído	13
2.3	<i>Beamforming</i> distribuído versus centralizado	14
2.4	Revisão da literatura	14
2.5	Sincronização em <i>beamforming</i> distribuído	16
2.6	Sincronismo em Malha Fechada	20
2.6.1	Sincronismo com Realimentação Integral (<i>Full Feedback</i>)	20
2.6.2	Sincronismo com Realimentação Parcial(<i>One-bit Feedback</i>)	22
2.7	Sincronismo em Malha Aberta	25
2.7.1	Método <i>Round Trip</i>	25

2.7.2	Método <i>Time-Slotted Round Trip</i>	27
2.8	Conclusão	31
3	Método Proposto	32
3.1	Modelo simplificado do sistema	33
3.2	Método Proposto - Princípio básico	34
3.3	Método Proposto (Inversão Remota de Fase - IRF)	38
3.3.1	Inversão Remota de Fase (IRF)	40
3.3.2	Minimização de Latência	41
3.4	Conclusão	44
4	Testes e Resultados	45
4.1	Resultados obtidos por meio de Simulação	46
4.1.1	Princípio básico	46
4.1.2	Aplicação	48
4.2	Resultados Experimentais	49
4.2.1	Inversão de Fase (IRF) com fase conhecida	50
4.2.2	Inversão remota de fase (IRF)	53
4.2.3	Time-slotted Round Trip (RT)	57
4.2.4	One Bit Feedback (OBFB)	58
4.2.5	Comparações	59
4.3	Conclusão	63
5	Conclusão	64
5.1	Considerações e Sugestão para estudos futuros	66
A	Implementação	68
A.1	Plataforma Experimental SDR	68
A.1.1	<i>Software</i>	68
A.1.2	<i>Hardware</i>	69
A.2	Arcabouço	70
A.3	Implementação dos Blocos Principais	72

A.3.1	Deslocador de Fase (<i>Phase_shifter</i>)	72
A.3.2	PLL Digital (<i>DPLL_hold</i>)	73
A.3.3	Controlador (FSM)	74
A.4	Implementação dos Interrogadores	76
A.5	Implementação dos Transceptores	79
A.6	Conclusão	86
B	Método Alternativo	87
B.1	Inversão de Fase baseada em Duplo Tom (IFDT)	87
B.2	Possível implementação do método IFDT	90
B.3	Resultados obtidos por meio de simulação (IFDT)	92
B.4	Comentários	93
	Referências bibliográficas	96

Capítulo 1

Introdução

A crescente demanda por sistemas de monitoramento e controle de ambientes e equipamentos tem gerado interesse contínuo na área de Redes de Sensores Sem Fio (RSSF) [1, 2]. O desenvolvimento e aperfeiçoamento de tais redes é um passo importante em direção à realização dos conceitos de Ambientes Inteligentes e Internet das Coisas, que podem modificar a maneira como o ser humano interage com o mundo ao seu redor [3]. Essas redes são projetadas para monitorar variáveis relativas ao meio físico, e em alguns casos, realizar ações de controle sobre o mesmo. Devido a suas dimensões reduzidas, o *hardware* utilizado em RSSF está submetido a restrições severas de energia e processamento. Uma forma de superar as restrições com recursos limitados é a implementação dessas redes na forma de sistemas colaborativos para que possam realizar suas tarefas [4, 5].

1.1 Redes de Sensores Sem Fio (RSSF)

Uma rede de sensores sem fio (RSSF) é um tipo especial de rede de computadores formada por grupo de dispositivos com capacidade de sensoriamento, processamento e comunicação, chamados nós sensores, interconectados por uma rede de comunicação sem fio. Cada nó sensor deve adquirir dados, processá-los e enviá-los para um nó de destino [6]. O hardware dos nós sensores é composto de diversos subsistemas: energia, sensoriamento, processamento e comunicação [4, 7, 8]. A estrutura de uma RSSF típica está ilustrada na Figura 1.1.

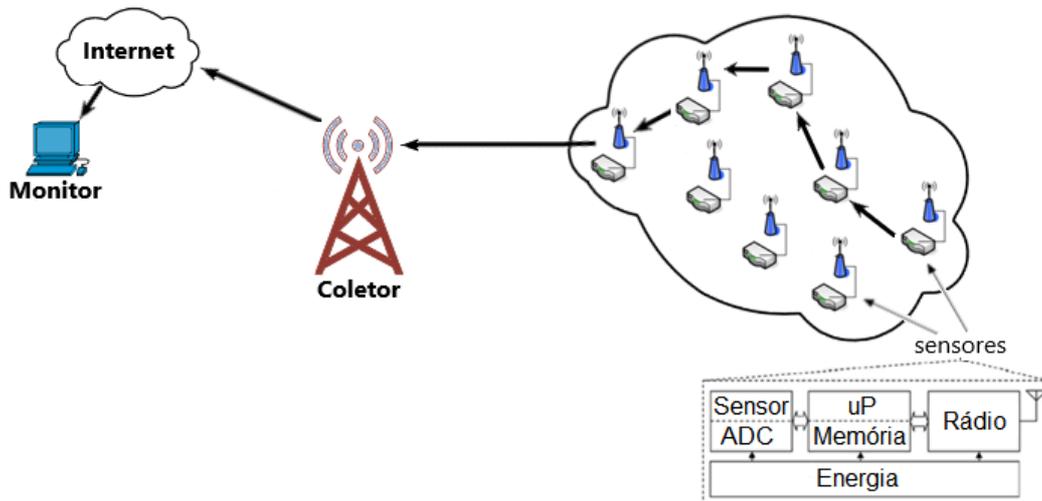


Figura 1.1: Representação de uma Rede de Sensores Sem Fio típica.

Na maioria das aplicações, os nós sensores estão sob restrições severas de espaço, peso, processamento e, principalmente, consumo de energia, uma vez que a transmissão dos dados requer muito mais energia que o sensoriamento e o processamento dos dados [10]. A transmissão direta a partir de um nó sensor até uma base distante pode requerer alta potência, o que não é desejável devido às restrições já citadas. Nesse caso, adota-se o modo de múltiplos saltos no qual nós intermediários retransmitem a informação até que ela chegue ao destino, situação essa, ilustrada na Figura 1.2, em que o nó A faz uso dos nós repetidores B, C, D e E para entregar uma mensagem ao coletor.

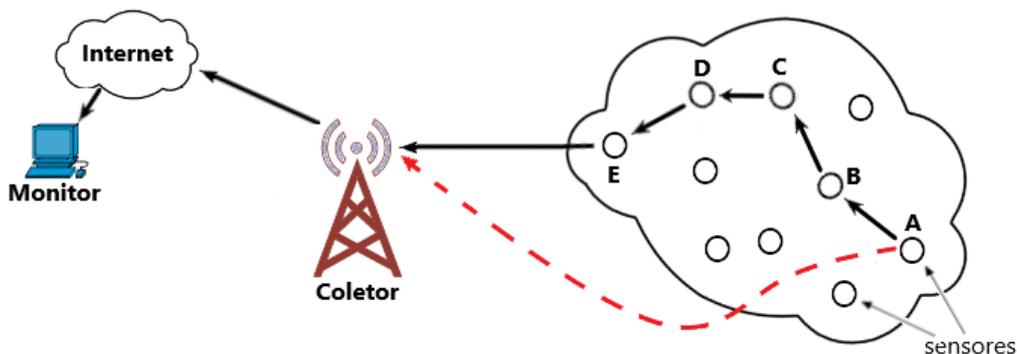


Figura 1.2: Diagrama representativo de uma RSSF com colaboração via retransmissão.

Entretanto, em algumas situações não existe nó retransmissor capaz de se comunicar

diretamente com o coletor, por exemplo, devido à falhas em que o último nó de uma cadeia de repetição deixa de funcionar, tornando a rede inoperante, conforme ilustrado na Figura 1.3.

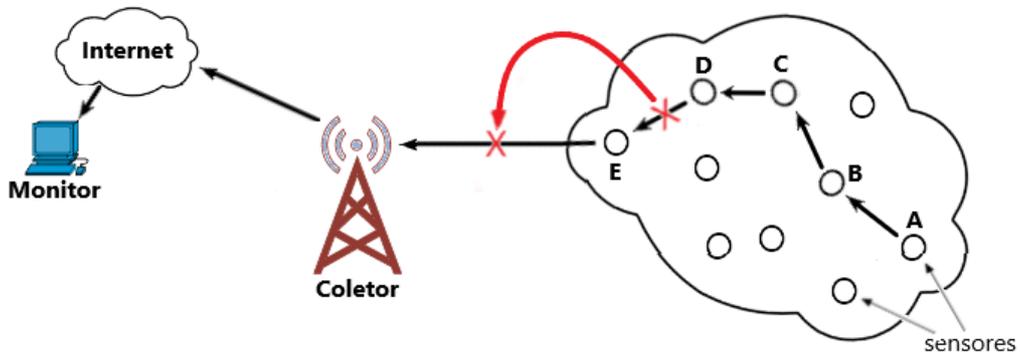


Figura 1.3: RSSF inoperante por falha no último repetidor da rede.

Situação similar é encontrada em aplicações especiais em que o coletor se encontra fora do alcance de transmissão de todos os sensores da rede, ou ainda como uma aplicação ilustrada na Figura 1.4, na qual um drone adquire informações de um conjunto de sensores em uma dada região (situação conhecida como *reach-back problem*) [9].

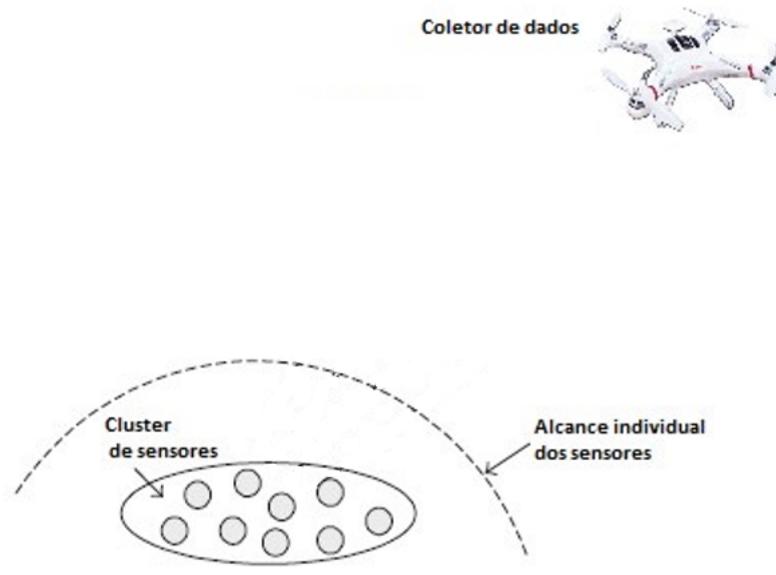


Figura 1.4: RSSF fora do alcance do coletor

Nesses casos, os nós-sensores poderiam operar em modo de transmissão colaborativa, cooperando entre si e se comportando como um arranjo virtual de antenas pela utilização de uma técnica conhecida como *beamforming* distribuído (*distributed beamforming*), com o objetivo de aumentar o alcance da rede [11], conforme ilustrado na Figura 1.5.

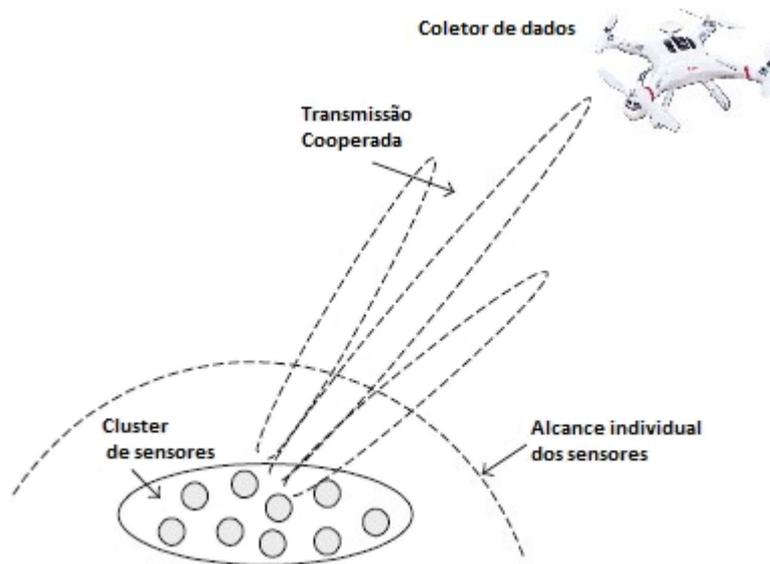


Figura 1.5: RSSF com colaboração via *beamforming* distribuído

1.2 *Beamforming*

As ondas de rádio ao se propagarem estão sujeitas a reflexões que provocam alterações na sua amplitude e caminho percorrido, ocasionando variações na potência do sinal recebido, fenômeno denominado de desvanecimento (do inglês *fading*) [12]. Na Figura 1.6 está representado um enlace de rádio terrestre em que existem dois caminhos de propagação entre o transmissor (TX) e o receptor (RX), um direto e um refletido pelo solo. Assumindo um coeficiente de reflexão no solo, $\rho = e^{j\phi}$, a intensidade do campo magnético, E , no receptor, será modificada por um fator $F = 1 + \rho e^{j\phi} e^{j2\pi(d_2 - d_1)/\lambda}$, com d_1 e d_2 representando os comprimentos dos caminhos direto e refletido, respectivamente e $2\pi(d_2 - d_1)/\lambda$ representando a diferença de fase entre os sinais que percorrem esses caminhos.

O resultado da recepção combinada dos dois sinais (direto e refletido) pode ser visto na

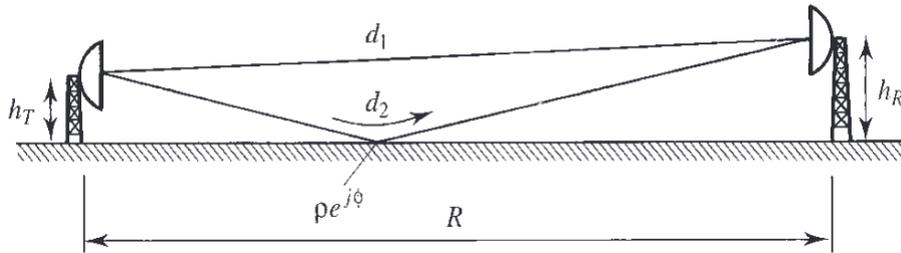


Figura 1.6: Representação de um enlace de rádio com multi-percurso. Fonte: reproduzido de [12].

Figura 1.7, que representa a variação da potência recebida devido ao efeito de multi-percurso. Dependendo da diferença de fase existente entre o sinal direto e o sinal refletido, a combinação desses sinais pode se manifestar como uma interferência construtiva ou destrutiva. Na Figura é possível notar que os picos correspondentes aos pontos de interferência construtiva estão 6 dB acima do valor correspondente ao nível no espaço livre e os pontos de interferência destrutiva estão 20 dB abaixo desse mesmo nível.

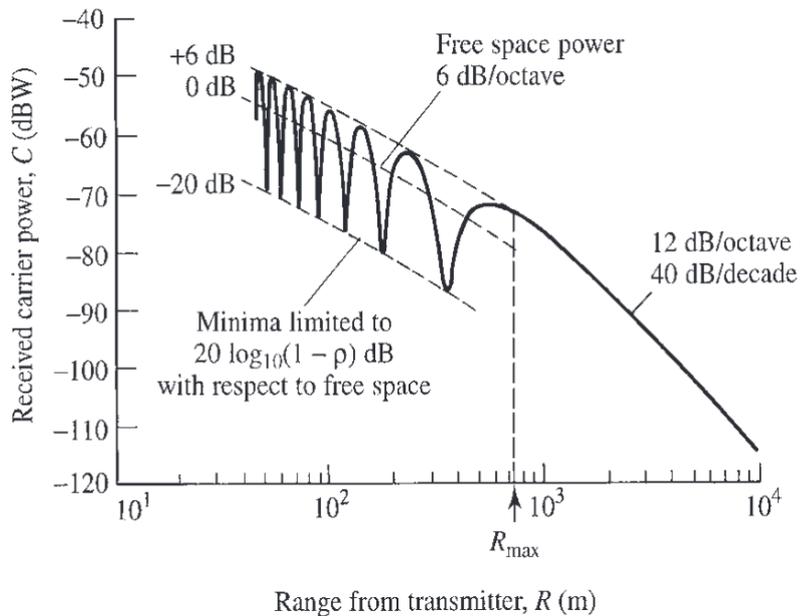


Figura 1.7: Variações de potência causadas por multi-percurso. Fonte: reproduzido de [12].

Se o ponto de reflexão do sinal, na Figura 1.6, for considerado como uma segunda fonte de sinal, pode-se inferir que o controle de fase desse segundo sinal poderia permitir o con-

trole da amplitude do sinal resultante no receptor.

A transmissão de um sinal por múltiplas antenas provoca efeito semelhante, como ilustrado na Figura 1.8, em que os sinais no receptor se combinam de forma construtiva ou destrutiva, dependendo da fase de cada sinal. O ajuste preciso de frequência e fase desses sinais permite controlar a forma da combinação no receptor. Esse é o princípio fundamental em que se baseia a técnica de *beamforming*, em que dois ou mais transmissores alinham as fases de suas portadoras de forma a provocar uma interferência construtiva no receptor, com o objetivo de aumentar nível de sinal e, como consequência, melhorar a relação sinal-ruído (SNR).

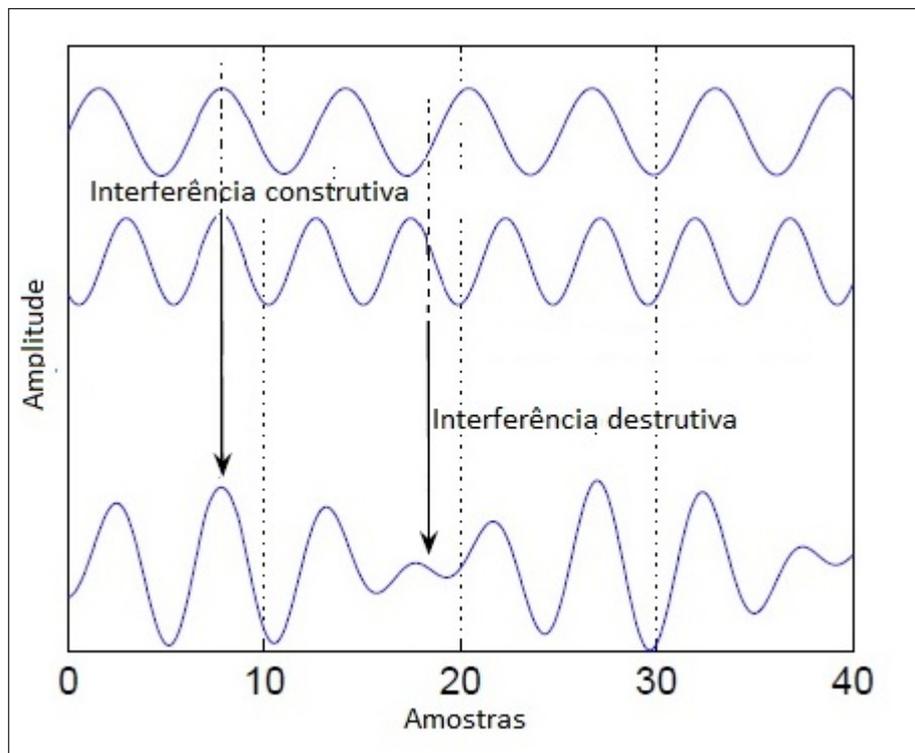


Figura 1.8: Interferências construtivas e destrutivas.

Tradicionalmente, *beamforming* tem sido implementado usando arranjos de antenas que são sistemas centralizados, conhecidos como *Phased Arrays*, com cada antena recebendo um sinal de RF derivado de um único oscilador local. A fase de cada derivação é função da direção desejada para o feixe de transmissão ou de recepção [13], como ilustrado de forma simplificada na Figura 1.9, em que o feixe de irradiação aparece inclinado (direcionado) em

relação ao eixo perpendicular ao plano das antenas devido ao deslocamento de fase, ($\delta\phi$), adicionado ao sinal da antena *Ant 1*. Um receptor posicionado em um ponto ao longo do feixe de irradiação pode observar um aumento de $20\log_{10}(2)\simeq 6$ dB na potência do sinal quando comparado com um sinal transmitido por apenas uma antena [14].

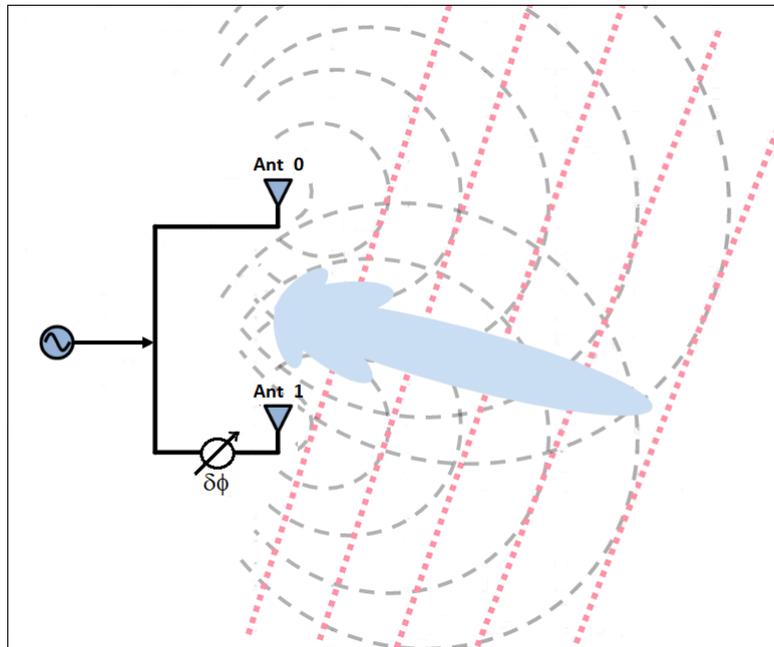
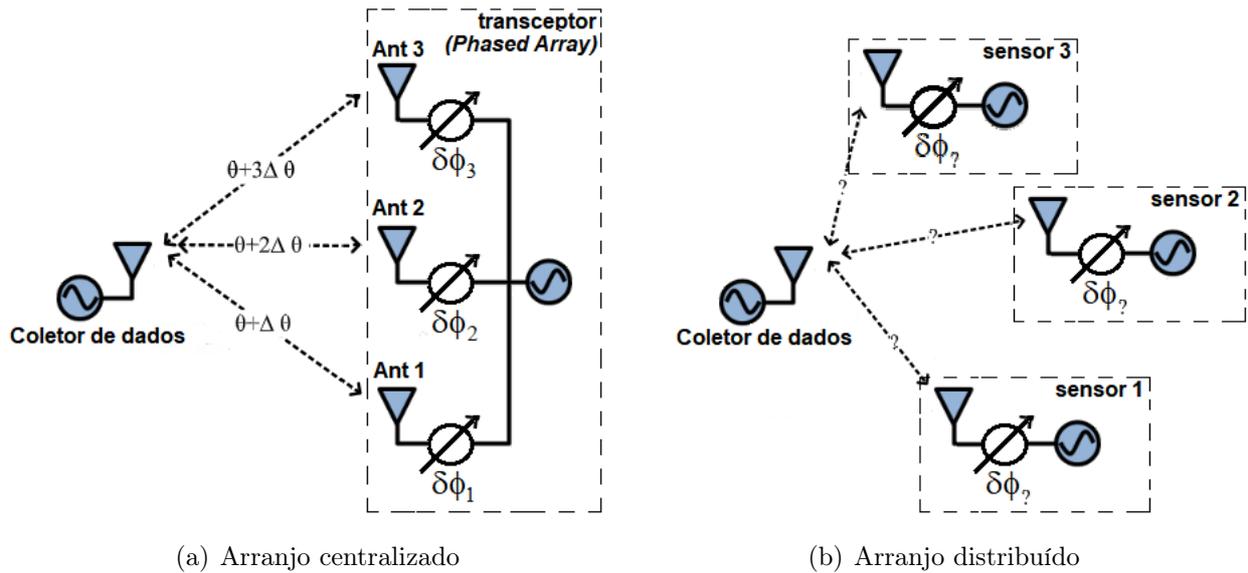


Figura 1.9: Direcionamento de feixe em Arranjo de Antenas (*Phased Array*).

Diferentemente dos *Phased Arrays*, as redes de sensores são sistemas distribuídos, e, portanto, seus elementos não compartilham um sinal comum que possa ser utilizado como referência mútua. Como mencionado por R. Mudumbai et al em [15], isso torna a implementação de *beamforming* uma tarefa desafiadora, já que os elementos da rede precisam sincronizar não apenas as frequências de suas portadoras, mas também suas fases, uma vez que o oscilador local de cada elemento da rede apresenta um deslocamento de fase independente e, possivelmente, desconhecido. Essa diferença fundamental entre *beamforming* centralizado e *beamforming* distribuído está destacada de forma simplificada na Figura 1.10.



(a) Arranjo centralizado

(b) Arranjo distribuído

Figura 1.10: Comparação entre a) *Beamforming* centralizado e b) *Beamforming* distribuído

1.3 Motivação

As principais motivações para essa tese são:

- Os métodos existentes baseados em sistemas de malha fechada são interativos e apresentam tempo de convergência variável, o que leva a uma indeterminação com relação ao tempo de resposta do sistema;
- Os métodos baseados em malha aberta necessitam de sincronismo explícito entre os nós escravos exigindo, continuamente, múltiplas interações entre eles, com conseqüente consumo de energia adicional e maior complexidade no processamento;
- Defende-se a hipótese de que os métodos de *beamforming* utilizados em arranjos refletores ou retrodiretivos, baseados na inversão da fase de chegada do sinal, podem ser adaptados para uma abordagem descentralizada

1.4 Objetivos

O objetivo geral dessa tese é apresentar um método para sincronismo distribuído, em malha aberta, visando transmissão cooperativa em redes de sensores utilizando *beamforming* distribuído.

1.4.1 Objetivos específicos

Os objetivos específicos dessa tese são:

- Compreender os mecanismos inerentes ao processo de *beamforming* distribuído;
- Identificar eventuais oportunidades de adaptação existentes nos arranjos centralizados;
- Desenvolver um método distribuído não-interativo para estimação e controle de fase;
- Aplicar esse método na implementação de *beamforming* distribuído.

1.5 Contribuições

Neste tópico, são apresentadas as contribuições resultantes dessa tese.

1.5.1 Principais contribuições - *beamforming* distribuído

As principais contribuições relacionadas diretamente com *beamforming* distribuído são:

- Método para alinhamento de portadora baseado na inversão de fase de um sinal piloto.

O método se baseia na inversão da fase de um sinal piloto enviado pelo interrogador para os sensores. Utilizando um protocolo Mestre-Escravo, cada sensor envia sua estimativa de fase de chegada do sinal piloto para o Mestre (um elemento da rede). O Mestre, inverte e devolve essa estimativa invertida para os sensores que a utilizam como fator compensador de fase para uma transmissão de retorno para o interrogador. Essa inversão de fase cancela os efeitos indesejados das fases iniciais dos osciladores locais de cada sensor, permitindo uma transmissão com recepção coerente no ponto de interrogação.

- Aplicação desse método na implementação de *beamforming* distribuído.

1.5.2 Outras contribuições - plataforma SDR

As contribuições relacionadas à implementação dos métodos em SDR são:

- Desenvolvimento de funções modulares adicionais ao ambiente *GNU-Radio*;
- Implementação de um arcabouço básico de *software* para experimentação de técnicas de *beamforming* distribuído no ambiente *GNU-Radio*.

1.5.3 Publicações

- Angilberto M. F. Sobrinho, Edmar C. Gurjão, José S. da R. Neto, “Distributed Beamforming Based on Dual-Frequency PDoA for Wireless Sensor Network”, IEEE International Symposium on Circuits and Systems (ISCAS), 2019.
- Angilberto M. F. Sobrinho, Edmar C. Gurjão, José S. da R. Neto, “Beamforming distribuído baseado na inversão remota de fase da portadora: Uma prova de conceito em GNU-Radio/USRP”, em XXXVIII SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS - SBrT 2020, FLORIANÓPOLIS, SC, 2020
- Angilberto M. F. Sobrinho, Edmar C. Gurjão, José S. da R. Neto, “Carrier Phase Alignment for Distributed Beamforming”, Journal of Communication and Information Systems (JCIS), 2021 (Submetido).

1.6 Estrutura do Documento

Este documento é composto de cinco capítulos e dois apêndices, e está estruturado da seguinte forma. No Capítulo 1, introdutório, contextualiza-se o tema da tese. No Capítulo 2, apresenta-se uma revisão da literatura dos principais trabalhos relacionados ao tema e faz-se uma descrição de dois dos principais métodos conhecidos. No Capítulo 3, é apresentado o método proposto e no Capítulo 4 são apresentados os resultados obtidos. No Capítulo 5, são apresentadas as considerações finais e a conclusão bem como sugestão para trabalhos futuros. No Apêndice A são descritas as implementações do método proposto, bem como do arcabouço de *software* sobre a plataforma de *software* livre *GNU-Radio* que dá suporte ao desenvolvimento do sistema. Por fim, no Apêndice B é apresentado um método alternativo,

ainda preliminar, para alinhamento de fase baseado em duplo tom de treinamento, com o objetivo de diminuir a latência e torna-la independente do número de sensores.

Capítulo 2

Revisão da Literatura

Neste capítulo, após a definição do problema, apresenta-se uma revisão da literatura relacionada a *beamforming* distribuído. São apresentadas as abordagens baseadas em mecanismos de malha fechada e malha aberta, e conclui-se com a descrição de dois esquemas representativos das duas abordagens.

2.1 Definição do Problema

O problema investigado nessa tese é similar ao problema da soma de dois sinais senoidais com mesma frequência mas com fases diferentes e que pode ser descrito como:

Se dois sinais, $S_0 = \cos(\omega t + \phi_0)$ e $S_1 = \cos(\omega t + \phi_1)$, com frequência angular w e fases iniciais ϕ_0 e ϕ_1 respectivamente, são transmitidos simultaneamente por dois transmissores distintos, o sinal resultante em um ponto distante de recepção é dado por

$$R(t) = S_0(t) + S_1(t) = \cos(\omega t + \gamma_0) + \cos(\omega t + \gamma_1), \quad (2.1)$$

com $\gamma_i = \phi_i + \theta_i$, em que θ_i é igual ao deslocamento de fase devido ao percurso entre o transmissor i e o ponto de recepção.

Fazendo uso de identidades trigonométricas, essa expressão pode ser reescrita como

$$R(t) = 2\cos\left(\omega t + \frac{\gamma_0 + \gamma_1}{2}\right)\cos\left(\frac{\gamma_0 - \gamma_1}{2}\right), \quad (2.2)$$

e o termo $\cos(\frac{\gamma_0 - \gamma_1}{2})$ pode ser interpretado como um fator de coerência entre os sinais. Para qualquer $\gamma_0 \neq \gamma_1$, o termo $\cos(\frac{\gamma_0 - \gamma_1}{2})$ é menor que 1, com conseqüente redução no nível do sinal resultante. Quanto mais próximo de zero for a diferença entre γ_0 e γ_1 , maior será a interferência construtiva e maior será o nível do sinal resultante e viceversa. Portanto, toda e qualquer abordagem que vise a implementação de *beamforming* distribuído sofrerá impacto causado pela diferença de fase (desconhecida) entre os elementos da rede.

Esse impacto pode ser minimizado pelo uso de técnicas de sincronização baseadas em sistemas de controle em malha aberta [16]-[21], ou em malha fechada [22]-[27]. Essas técnicas ainda são objeto de estudo e pesquisa.

2.2 *Beamforming* distribuído

Beamforming distribuído é uma técnica de transmissão na qual elementos sensores independentes e distribuídos geograficamente de forma aleatória, cooperam entre si para formar um arranjo virtual de antenas. Essa técnica baseia-se na utilização de múltiplos rádios independentes e com mesma frequência que transmitem simultaneamente a mesma informação utilizando portadoras cujas fases são alinhadas de forma a criar uma interferência construtiva no ponto de recepção [28]. Esta técnica tem por objetivo a maximização da energia irradiada em uma direção em particular, resultando em um aumento do ganho direcional, o que pode permitir maior alcance de transmissão, maior taxa de transmissão ou menor consumo de energia, dependendo da aplicação pretendida. Em um modelo proposto por H. Ochiai et al, em [29], é demonstrado que com uma potência fixa irradiada por cada elemento da rede, um sistema ideal de *beamforming* composto de N elementos resulta em um ganho de N^2 na potência recebida pelo destino [28] ou, dito de outra forma, a potência de transmissão pode ser reduzida por um fator de $1/N^2$ para um nível determinado de potência no receptor.

2.3 *Beamforming* distribuído versus centralizado

Pode-se dizer, de forma simplificada, que *Beamforming* é uma adaptação dos mecanismos abordados pela teoria tradicional de arranjo de antenas no que diz respeito à conformação direcional de feixes de radiação de energia.

As diferenças mais evidentes entre os dois sistemas (centralizados e distribuídos) estão listadas na Tabela 2.1.

Em geral, sistemas centralizados de *beamforming* são baseados em arranjos uniformes de antenas em que os elementos são posicionados de forma uniforme e preestabelecida. Em redes distribuídas de sensores, por outro lado, os elementos que formam o arranjo virtual de antenas (sensores) são organizados de forma aleatória e seu posicionamento não pode ser manipulado. Ainda, em sistemas centralizados os elementos do arranjo compartilham um mesmo processador e se utilizam de um oscilador, também comum a todos, que estabelece uma referência de tempo, fase e frequência para suas portadoras de comunicação. Essa referência comum não existe em sistemas distribuídos – cada sensor (elemento do arranjo virtual) deve ser capaz de obter sincronismo de fase, frequência e tempo, de forma independente.

Tabela 2.1: Diferenças entre *beamforming* distribuído e centralizado

Aspecto	<i>Centralizado</i>	<i>Distribuído</i>
Distribuição das antenas	uniforme	aleatória
Sincronização	controlador comum	controladores independentes
Osciladores	oscilador comum	osciladores independentes
Tamanho do arranjo	pequeno (alguns λ^*)	grande (dezenas ou centenas de λ^*)

λ^* = Comprimento de onda.

2.4 Revisão da literatura

Em 2009, Mudumbai et al [15], discutiram a evolução dos sistemas de *beamforming* distribuído durante a década passada a partir de trabalhos publicados em 2004, que estabeleceram as bases para essa tecnologia [30, 31] e apresentaram os principais desafios a serem enfrentados para a realização de sistemas práticos.

Com o nome de *Collaborative Beamforming*, Ochiai et al. [30] fazem uso da teoria de arranjo aleatório de antenas para investigar a conformação de feixes a partir do controle de diagramas de radiação das antenas, considerando que os elementos se encontram perfeitamente sincronizados em fase entre si.

Ao mesmo tempo, Barriac et al. [31], sem considerar os diagramas de radiação nem a geometria física do arranjo, se concentrou na viabilidade de sincronismo entre os elementos da rede visando coerência de fase.

Esses dois aspectos são discutidos de forma unificada, em 2011, por Uher et al. [32] em que a conformação de feixe é analisada considerando tanto a geometria do arranjo quanto os mecanismos para sincronismo de frequência e fase.

Em [33], Jayaprakasam et al., apresentam uma discussão bem completa sobre diversos aspectos da tecnologia de *beamforming* distribuído e comunicação colaborativa e fazem um levantamento atualizado da literatura existente, trazendo a discussão para um contexto mais moderno considerando as tendências atuais de comunicação incluindo a relação com tecnologia 5G e aplicações voltadas para redes de sensores.

Conforme discutido por J. Uher et al, em [32], a distribuição geográfica e o posicionamento dos sensores tem papel relevante na implementação de *beamforming* e, em [34], H. Ochiai et al, consideram uma rede cujos sensores foram distribuídos de três formas diferentes: distribuição aleatória, distribuição uniforme e distribuição gaussiana e demonstram que é possível alcançar ganho de N^2 em uma direção específica. Os resultados mostram que o lóbulo lateral de radiação se situa em torno de -12 dB para uma rede de $N = 16$ sensores e em torno de -24 dB para uma rede de $N = 256$ sensores.

A partir dos resultados de simulação, os autores também deduzem que a largura do feixe principal é inversamente proporcional ao número de sensores, N . Já em [35]-[37], Mohammed F. A. Ahmed et al, consideram uma rede cujos elementos são posicionados segundo uma distribuição gaussiana e concluem que o posicionamento uniforme apresenta um feixe de radiação mais estreito, mas o posicionamento gaussiano minimiza os lóbulos laterais de radiação.

Por outro lado, o foco inicial dos trabalhos citados por G. Barriac [31] e R. Mudumbai [21] foi a possibilidade de formar um feixe de radiação em uma direção específica, sem considerações relacionadas com a geometria do arranjo. Nem a localização dos sensores

nem a direção do feixe são explicitamente conhecidas e o objetivo principal do sistema é permitir que os elementos distribuídos identifiquem a direção de interesse com o mínimo de informação disponibilizada pelo receptor distante e que, subsequentemente, ajustem as fases iniciais de suas transmissões para formar um feixe com máxima potência naquela direção.

As pesquisas baseadas em análise de diagramas de radiação apresentadas por H. Ochiai et al [30, 34], frequentemente consideram que os sistemas se encontram em perfeita sincronia de fase e frequência, quando, na realidade, sempre existem diferenças de temporização, frequência e fase entre os sinais dos elementos que compõem o arranjo. Para que os sinais se combinem de forma coerente no receptor distante, é necessário levar em conta e compensar essas diferenças. Portanto, *beamforming* distribuído só é alcançado quando se consegue sincronizar de forma eficiente os sinais dos diferentes elementos da rede. Como mencionado por R. Mudumbai et al, em [15], esse assunto é um dos maiores desafios que se apresenta na implementação de sistemas reais de *beamforming* distribuídos.

2.5 Sincronização em *beamforming* distribuído

De uma forma geral, a maioria dos métodos de sincronização para *beamforming* distribuídos encontrados na literatura podem ser classificados em duas categorias gerais: métodos em malha aberta e métodos em malha fechada. Os métodos em malha fechada podem ser subdivididos em sistemas com realimentação parcial e sistemas com realimentação integral. Similarmente, os sistemas baseados em malha aberta podem ser subdivididos em sistemas de comunicação cega e sistemas com comunicação entre elementos.

Uma classificação geral dos principais métodos de *beamforming* distribuído pode ser vista na Figura 2.1, com o tracejado indica que existem variações daquele método.

- Duas abordagens básicas:

- **Malha fechada:**

Os sistemas de malha fechada [22]-[27], se baseiam em métodos iterativos nos quais o receptor distante (interrogador) participa ativamente do processo de convergência de fase. A intensidade relativa do sinal recebido pelo interrogador é informada (realimentada) para os elementos sensores que efetuam ajustes nas

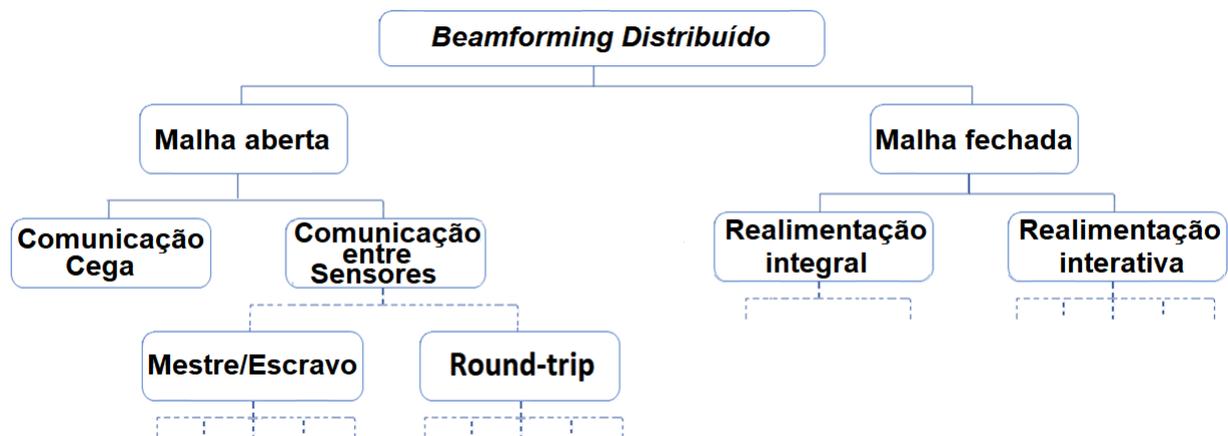


Figura 2.1: Principais métodos de *beamforming* distribuído.

fases de seus osciladores locais.

– **Malha aberta:**

Nos sistemas de malha aberta [16]-[21], a sincronização entre os elementos é alcançada com o mínimo de intervenção do interrogador.

O interrogador apenas transmite um sinal piloto de treinamento para os sensores que, a partir desse sinal piloto, utilizam múltiplos PLLs (*phase locked loops*) e trocam mensagens entre si para sincronizar seus transmissores.

Realizações experimentais representativas de métodos em malha fechada e malha aberta são apresentadas por M. M. U. Rahman et al [38] e D. R. Brown et al [39], respectivamente.

Em [40] e [21], G. Barriac et al, propõem e analisam um método para sincronismo baseado em um protocolo mestre-escravo utilizando uma arquitetura em estrela. Nestes trabalhos, os autores demonstram a convergência do algoritmo e comprovam que a relação sinal/ruído aumenta com o número de sensores. Ganhos de até 70% são alcançados mesmo com erros de fase em torno de 30° .

Em [41,55], U. Madhow et al, apresentam um método em malha fechada que utiliza um processo semelhante ao algoritmo gradiente ascendente, no qual a função de custo é o nível de sinal recebido pelo interrogador, que é realimentada de forma quantizada para os sensores. Diversos métodos utilizam uma abordagem semelhante e são conhecidos genericamente como

métodos *bit feedback*. A escalabilidade do método é discutida em [55].

Algoritmos em malha fechada (*bit feedback*) são estudados por Stephan Sigg et al, em [42, 43], utilizando distribuição uniforme e gaussiana para perturbações de fase nos sensores. Os autores sugerem que o tempo de convergência dos algoritmos pode ser reduzido pelo agrupamento de sensores em blocos hierárquicos.

Em [44], M. O. Pun et al propõem um método oportunista como alternativa ao método clássico de *bit feedback* em que, ao invés de provocar perburbação na fase dos osciladores locais, um conjunto de sensores que naturalmente estão em quasi-coerência são escolhidos e selecionados para participar do processo de *beamforming*. Esse método pode ser aplicado em cenários em que não é possível ajustar a fase do oscilador local de forma controlada e se baseia em uma seleção centralizada.

Outra extensão do método *bit feedback* foi proposta por S. Song et al, em [45]. A realimentação quantizada de um *bit* é estendida para três *bits* por Wayes Tushar et al, em [46]. Um *bit* é usado da mesma forma que em [41], e dois *bits* adicionais são usados para indicar a direção do movimento do sensor em relação ao interrogador. Os resultados mostram que a potência do sinal recebido é maior que a potência apresentada no método clássico e aumenta com o quadrado do número de sensores. Em [47], Gubong Lim et al, dividiram os sensores em dois grupos e o algoritmo foi aplicada a cada um dos grupos.

Um exemplo de sistema em malha aberta que se utiliza de comunicação entre os sensores para alcançar sincronização é apresentado por H. Ochiai et al, em [28], em que um elemento da rede é responsável pela estimação inicial das diferenças de fase. Em [48], [38] e [31], F. Quitin, R. Mudumbai e U. Madhow, apresentam sistemas com arquiteturas mestre-escravo em que um elemento é escolhido como mestre e transmite um sinal contínuo de treinamento como referência para os outros elementos (escravos).

Um método, em malha aberta, baseado no tempo de ida-e-volta do sinal (*Round-Trip*) foi proposto por M. M. Rahman et al, em [49], composto de um interrogador e dois sensores. O sincronismo é alcançado pela utilização de um sinal piloto enviado pelo interrogador que é retransmitido por cada sensor em um circuito em forma de anel em duas direções diferentes, do interrogador para o primeiro sensor e do primeiro sensor para o segundo. Simultaneamente, o sinal piloto recebido do interrogador pelo segundo sensor é retransmitido para o primeiro sensor. Em [50], D. Richard Brown III et al, apresentaram uma proposta expan-

dindo esse método para aplicações com N sensores e apresentaram uma análise considerando cenários como multi-percurso e levam em consideração aspectos de variação temporal do canal de comunicação.

Em [51], R. D. Preuss et al, apresentam um método denominado TW (*Two-Way Source Synchronization*), para facilitar a formação de *beamforming* de forma retro-diretiva. Um sinal piloto é transmitido de sensor a sensor em uma direção e, depois, é retransmitido novamente na direção inversa. Cada sensor estima o deslocamento de fase sofrido pelo sinal piloto em uma direção e o combina com o deslocamento de fase apresentado na outra direção. O protocolo é similar ao método *Round-trip*, com a diferença que o interrogador não participa do processo.

Em [52]-[53], Aggelos Bletsas et al, apresentam um método de comunicação cega com a sincronização de fase sendo obtida pela repetição contínua da transmissão da mensagem considerando que existe uma probabilidade das fases dos sinais se alinharem independente de controle explícito. Este é um método muito particular, pois considera que a natureza periódica dos deslocamento de fase dos osciladores locais permite, mesmo que de forma intermitente, uma coerência de fase no receptor do interrogador. Uma análise estatística do método, efetuada pelos mesmos autores, em [54], mostra que o método é limitado a pequenas redes e se aplica apenas a situações de emergência.

Em [75], H. Yan et al, apresentam um método para sincronismo de frequência e de temporização, sem considerar o ajuste de fase, em que um sinal transmitido por um elemento denominado mestre, é utilizado pelos elementos denominados escravos para estimar a frequência e corrigir seus osciladores locais. O método explora o uso de um preâmbulo baseado em uma sequência periódica Zadoff-Chu.

J. C. Merlano-Duncan et al, em [76] propõem um método baseado em comunicação bidirecional com um sinal piloto composto de múltiplas frequência para sincronismo remote de portadora e sugerem seu uso em aplicações de comunicação entre satélites.

Em [77], Mghabghab, S. et al, demonstram um sistema em malha aberta operando na frequência de 1.5 GHz. O sistema é composto de dois elementos que apresentam movimentos relativos entre si. Um sinal piloto baseado em duplo tom é utilizado sincronismo de frequência e, adicionalmente, utilizado para estimativas da distância entre os elementos. Cada elemento contém um receptor adicional que extrai o sinal de referência modulado na

diferença de frequência entre os dois tons do sinal piloto, e o utiliza como referência para os PLLs internos.

S. Prager et al, em [78] apresentam um algoritmo para sincronização por consenso, baseado em uma arquitetura mestre-escravo e que utiliza o sinal de um GPS para ajustes iniciais e refinamento por software.

2.6 Sincronismo em Malha Fechada

Os métodos de sincronismo em malha fechada são encontrados na forma de realimentação integral e realimentação parcial ou quantizada.

Em [55], R. Mudumbai et al descrevem um método representativo da abordagem em malha fechada e em [56] esse método é otimizado e apresentado como *One-bit Feedback Synchronization* que utiliza apenas um *bit* como indicador para o ajuste iterativo das fases das portadoras do sensores.

2.6.1 Sincronismo com Realimentação Integral (*Full Feedback*)

Em [16], S. Berger e A. Wittneben, utilizam uma abordagem Mestre-Escravo, na qual o interrogador é o mestre e os sensores são os escravos. O mestre tem a função de estimar o canal entre ele e os sensores e informa-los sobre essa estimativa.

Sincronismo com Realimentação Integral (*Paralela*)

Nesse método [16], a comunicação entre os escravos e o mestre se dá em dois passos e a diferença de fase entre o interrogador e cada sensor é corrigida seguindo o seguinte protocolo:

1. Para sincronismo de frequência, o mestre transmite em difusão, um sinal não modulado (*beacon*). Os escravos usam esse sinal para sincronizar seus PLLs;
2. Cada sensor transmite o *beacon* de volta para o mestre usando um método de acesso baseado em CDMA (*Code Division Multiple Access*);
3. Ao receber o *beacon* de volta, o mestre estima a diferença de fase vista por cada sensor em relação ao *beacon* original. Uma mensagem contendo as estimativas de canal é

enviada para os sensores via CDMA;

4. Cada sensor identifica e extrai sua estimativa de canal embutida na mensagem recebida;
5. Após corrigirem seus osciladores locais com as informações obtidas no passo 4, os escravos podem transmitir de forma cooperada para o destino.

Além da óbvia desvantagem da alta complexidade do esquema, devido ao uso de CDMA, esse método assume que o interrogador tem sensibilidade suficiente para detectar o sinal de cada sensor antes da formação do *beamforming*.

Em [80], S. Mohanti et al apresentam um sistema, relativamente complexo, baseado em malha fechada em que estimativas completas do canal são efetuadas pelo interrogador e informadas para os elementos da rede (elementos móveis, *drones*) que se sincronizam e transmitem de forma coerente para o interrogador instalado em uma base fixa em terra.

Sincronismo com Realimentação Integral(*Sequencial*)

Em [51], R. D. Preuss et al, apresentam um método em que a comunicação entre os escravos e o mestre, em uma rede com N sensores, se dá em $2N + 2$ passos e a diferença de fase entre o interrogador e cada sensor é corrigida seguindo o seguinte protocolo:

1. O mestre transmite um sinal não modulado (*beacon*) como referência. Os escravos usam esse sinal para sincronizar seus PLLs;
2. Cada sensor transmite, um de cada vez, o *beacon* de volta para o mestre;
3. Ao receber o *beacon* de volta, o mestre estima a diferença de fase vista por cada sensor em relação ao *beacon* original envia de volta uma mensagem contendo essa estimativa para o sensor correspondente;
4. Após corrigirem seus osciladores locais com as informações obtidas no passo 3, os escravos podem transmitir de forma cooperada para o destino.

Apesar de sua simplicidade e baixa complexidade, esse método apresenta a mesma desvantagem existente no método com realimentação integral paralela, que é assumir que o interrogador tem sensibilidade suficiente para detectar o sinal de cada sensor antes da formação do *beamforming*.

2.6.2 Sincronismo com Realimentação Parcial (*One-bit Feedback*)

Uma alternativa ao esquema de realimentação integral foi proposta por R. Mudumbai et al, em [22] e I. Thibault et al, em [23], usando apenas um *bit* de realimentação do interrogador para os sensores. Neste esquema, em vez de estimativas de canal, o interrogador apenas orienta os escravos para que provoquem pequenas rotações aleatórias de fase em suas portadoras e os informa se essas rotações contribuíram ou não para o nível desejado de coerência de fase.

Esse processo repetitivo leva, eventualmente, a uma convergência de fase dos sinais transmitidos pelos sensores.

Seu funcionamento está representado esquematicamente na Figura 2.2.

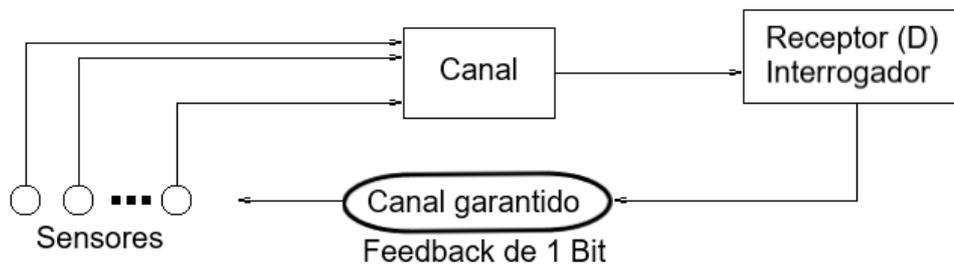


Figura 2.2: Esquema simplificado para o método *One bit Feedback*.

Considere uma rede composta de N sensores que transmitem cooperativamente uma mensagem comum, em banda-base complexa, $m(t)$ para um receptor distante.

Os sensores S_i estão sincronizados em frequência com uma portadora ω_c , mas cada sensor apresenta uma fase inicial diferente ϕ_i . Representando o sinal de RF transmitido por S_i

como $s_i(t)$ e a diferença de fase causada pelo canal entre S_i e o ponto de destino como θ_i , $s_i(t)$ pode ser escrito como

$$s_i(t) = \Re(m(t)e^{j(\omega_c t + \phi_i)}), \quad (2.3)$$

com $\Re(x)$ representando a parte real de x .

O sinal recebido no ponto remoto pode escrito como

$$r(t) = \sum_i^N \Re(m(t)e^{j(\omega_c t + \phi_i - \theta_i)}). \quad (2.4)$$

Representando a fase do sinal recebido de S_i como $\gamma_i = \phi_i - \theta_i$, a potência do sinal recebido, segundo R. Mudumbai et al, em [56], é dada por

$$S_r(t) = \left| \sum_i^N (e^{j\gamma_i}) \right|^2. \quad (2.5)$$

A potência do sinal recebido é maximizada quando os sinais se combinam de forma coerente no receptor, ou seja, quando $\gamma_i = \text{constante}$. Como não há controle direto sobre ϕ_i nem sobre θ_i , esse algoritmo está baseado na ideia de introduzir uma pequena rotação de fase, Ψ_i , no sinal em banda base para forçar a coerência de fase desejada, de modo que agora $\gamma_i = \phi_i - \theta_i + \Psi_i$.

A cada ciclo, ou *time-slot*, $[n]$ com período T_c , cada sensor efetua uma perturbação aleatória de fase na fase atual de sua portadora. O receptor estima a intensidade do sinal recebido e transmite para os sensores um *bit* de informação indicando se o nível de sinal é maior ou menor que nível de sinal do ciclo anterior $[n - 1]$.

Se o nível estiver maior, os sensores mantêm a última rotação de fase efetuada e seguem para o próximo ciclo. Se o nível de sinal estiver menor, os sensores descartam a última rotação de fase efetuada e retornam a fase para o valor definido no ciclo anterior antes de seguirem para o próximo ciclo.

O algoritmo pode ser sumarizado como (para cada ciclo):

1. Cada sensor efetua um ajuste independente em sua rotação de fase Ψ_i como,

$$\Psi_i[n] = \Psi_i[n - 1] + \delta\Psi, \quad (2.6)$$

com $\Psi_i[n]$ representando a rotação de fase no n -ésimo ciclo e $\delta\Psi$ é um valor aleatório.

2. Cada sensor aplica essa rotação de fase, Ψ_i , ao sinal e o transmite.
3. O receptor mede a potência média de $r(t)$ durante o n -ésimo ciclo, $P_r[n]$,

$$P_r[n] = \frac{1}{T_c} \int_{T_c} |r(t)|^2 dx \quad (2.7)$$

e compara com a medição do ciclo anterior $P_r[n - 1]$. Um sinal de realimentação com 1 *bit* de informação é gerado de acordo com

$$fb[n] = 1 \text{ se } P_r[n] > P_r[n - 1]; 0 \text{ caso contrário}, \quad (2.8)$$

e é transmitido para todos os sensores simultaneamente.

4. Os sensores analisam o *bit* recebido, $fb[n]$, e se $fb[n] > 0$, cada sensor mantém o ajuste de fase efetuado. Se não, o ajuste é descartado.

O processo continua, repetitivamente, até que o nível de sinal, no receptor, alcance um nível pre-determinado.

Uma aplicação desse método é apresentada por X. Fan et al, em [79], em que, após alcançar o nível de convergência desejado, o sistema passa a gerar pequenas perturbações na fase do sinal modulado causando grandes variações na amplitude do sinal para os receptores que não se encontram na região do interrogador, sendo útil em aplicações de comunicações sigilosas. Já em [81], os mesmos autores demonstram a aplicação do método *One bit Feedback* em um sistema de transmissão de energia sem fio.

A vantagem do método *One bit Feedback* é sua baixa complexidade de implementação e como desvantagem, o longo tempo de convergência, pois no início do processo de comunicação é muito provável que o sinal recebido seja muito fraco porque as portadoras dos escravos ainda não estão alinhadas. O interrogador, então, deve ser capaz de detectar variações muito pequenas nesse sinal para poder fornecer um *feedback* adequado para os sensores. Por isso, pode levar muito tempo para alcançar coerência já que cada interação necessita de um ciclo de comunicação ida-e-volta entre os sensores e o interrogador.

2.7 Sincronismo em Malha Aberta

Os métodos em malha aberta mais populares são baseados ou derivados do método *Round-Trip Carrier Synchronization* (RTCS), que utiliza topologia em anel e se baseia na equivalência dos atrasos de propagação de um sinal piloto, contínuo, originado no interrogador (D) e que é replicado de sensor a sensor, retornando ao ponto de origem (D) após “circular” o anel em duas direções opostas.

2.7.1 Método *Round Trip*

Considerando a parte hachureada da Figura 2.3 como exemplo para uma rede com 1 interrogador e 2 sensores, a ideia é que o sinal piloto circulando no sentido ($D \rightarrow S_0 \rightarrow S_1 \rightarrow D$) deve apresentar deslocamento total de fase igual ao deslocamento de fase apresentado no sentido ($D \rightarrow S_1 \rightarrow S_0 \rightarrow D$).

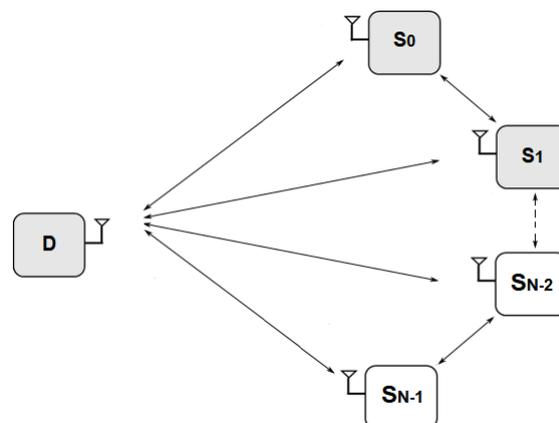


Figura 2.3: Diagrama de rede com 1 Interrogador e N sensores.

A proposta, interessante porém complexa do ponto de vista de implementação, é apresentada em [17], por D. Brown-III et al, em que os autores desenvolvem um método explícito para sincronismo entre as portadoras de dois transmissores e um receptor. O sistema proposto está representado na Figura 2.4.

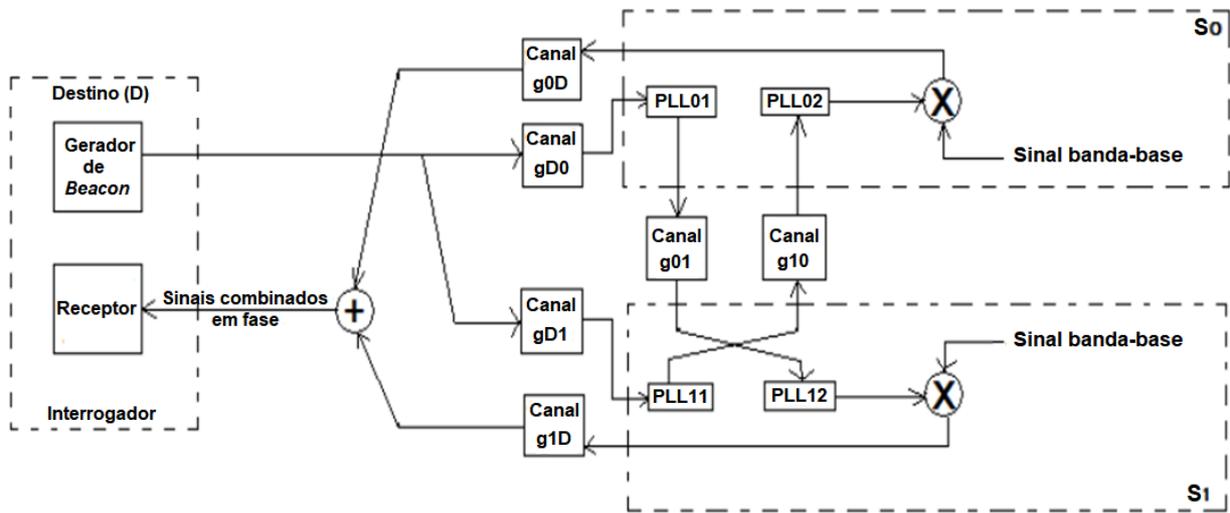


Figura 2.4: Esquema para sincronismo *Round Trip*. Fonte: Adaptado de [17].

Nesse esquema cada sensor dispõe de dois PLLs, operando de forma contínua. O interrogador (D) envia um sinal de *beacon* (portadora não modulada) para os sensores (S_0 e S_1). Cada sensor sincroniza seu PLL primário com esse sinal e o reenvia para o outro sensor. Simultaneamente, o PLL secundário de cada sensor é sincronizado com o sinal recebido do outro sensor e esse sinal é usado como portadora para transmitir de volta para o interrogador.

Observando-se a Figura 2.5, é possível perceber que os caminhos ($D \rightarrow S_0 \rightarrow S_1 \rightarrow D$) e ($D \rightarrow S_1 \rightarrow S_0 \rightarrow D$) são iguais, portanto os tempos de propagação também são iguais e com isso os sinais transmitidos pelos sensores chegam em fase no interrogador e se combinam de forma coerente.

Nesta abordagem foi abstraído o fato de que cada sensor tem apenas uma antena, mas deve se comunicar em três frequências diferentes, o que pode ser um grande problema, considerando-se a questão de espaço físico para a implementação de antenas multi-banda.

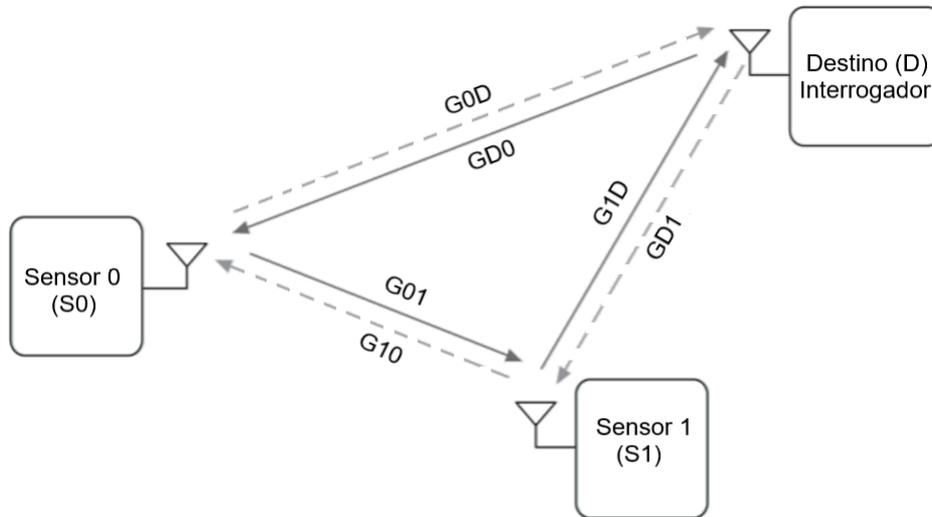


Figura 2.5: Caminhos de propagação no esquema *Round-Trip*.

2.7.2 Método *Time-Slotted Round Trip*

O método *Time-slotted round-trip carrier synchronization* (RT), foi introduzido por D. R. Brown III e H. Poorem, em [18], como uma evolução do método RTCS e uma implementação experimental foi apresentada por B. Zang, em [19].

Nesse método, utiliza-se um processo de *time slots* para sincronizar os nós, cuja representação espacial pode ser vista na Figura 2.5, e a representação temporal pode ser vista na Figura 2.6.

A circulação virtual do sinal pelo anel é implementada rastreando o sinal recebido pelo sensor com um PLL e, posteriormente, operando esse PLL em modo *holdover* para retransmitir uma extensão periódica do sinal recebido no intervalo de tempo anterior. A utilização de PLLs em diferentes configurações e, em especial, em modo de *holdover* no contexto de *beamforming* distribuído é discutida em detalhes por Ni. Min, em [20].

Em cada sensor são utilizados dois PLLs (PLL_1 e PLL_2), um para cada sentido de propagação.

A combinação coerente é alcançada no destino uma vez que o sinal recebido pelo interrogador é a soma dos sinais de retorno dos dois sensores depois da propagação por dois caminhos com deslocamentos de fase iguais, conforme descrito a seguir.

Em uma rede composta de N sensores, um ciclo completo do protocolo é realizado em

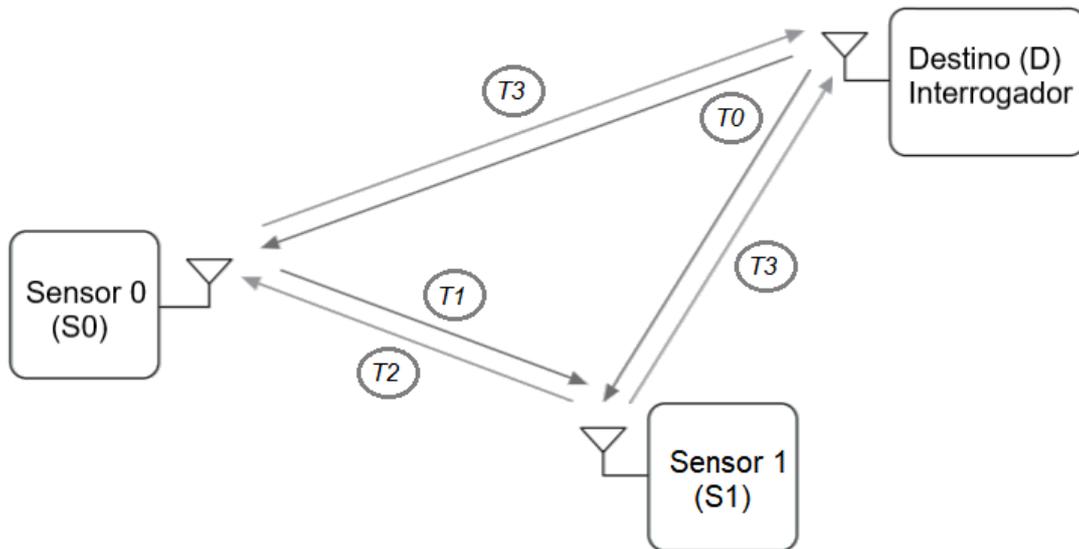


Figura 2.6: Sequência de propagação no método *Time-Slotted Round-Trip*.

$2N$ intervalos de tempo (*time-slots*). Para o exemplo de rede considerada tem-se quatro intervalos de tempo, TS_0 a TS_3 :

TS_0 : O interrogador transmite um sinal piloto que é recebido por cada sensor e utilizado para sincronizar o PLL_1 ;

TS_1 : O sensor S_0 transmite o sinal de seu PLL_1 para S_1 que o utiliza para sincronizar seu PLL_2 ;

TS_2 : O sensor S_1 transmite o sinal de seu PLL_1 para S_0 que o utiliza para sincronizar seu PLL_2 ;

TS_3 : Ambos os sensores transmitem, simultaneamente, os sinais de seus PLL_2 .

Essa sequência de eventos está sumarizada, graficamente, na Figura 2.7 e detalhada a seguir:



Figura 2.7: Sequência temporal do método *Time-Slotted Round-Trip*. Fonte: Adaptado de [18].

Considere que o sinal de referência transmitido pelo interrogador (D) durante o *time-slot* 0 pode ser representado por

$$tx_d(t) = \cos(\omega t + \phi_{tx_d}), \quad (2.9)$$

com ϕ_{tx_d} representando a fase inicial do oscilador local do transmissor e ω como a frequência angular de transmissão.

Cada nó sensor S_i ($i = 0, 1$), com fase inicial dos osciladores locais de transmissão e recepção iguais a ϕ_{tx_i} e ϕ_{rx_i} respectivamente, sincroniza seu PLL₁ ao sinal recebido, após a conversão para banda base

$$rx_i(t) = \cos(\omega_{bb}t + \phi_{tx_d} - \theta_i - \phi_{rx_i}), \quad (2.10)$$

com θ_i representando o deslocamento de fase experimentado pelo sinal devido ao atraso do percurso e ω_{bb} representando a frequência angular em banda base.

Durante o *time-slot* 1, o sensor S_0 transmite o sinal do seu PLL₁ para o sensor S_1 como

$$s_0(t) = \cos(\omega t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0}), \quad (2.11)$$

com $\phi_{\delta 0} = -\phi_{rx_0} + \phi_{tx_0}$.

O sensor S_1 recebe esse sinal, Equação (2.11), e, após conversão para banda base, sincroniza seu PLL₂ gerando

$$r_1(t) = \cos(\omega_{bb}t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0} - \theta_{ss} - \phi_{rx_1}), \quad (2.12)$$

com θ_{ss} representando o deslocamento de fase devido ao atraso do percurso ($S_0 \leftrightarrow S_1$).

Durante o *time-slot* 2, o sensor S_1 transmite o sinal do seu PLL₁ para o sensor S_0 como

$$s_1(t) = \cos(\omega t + \phi_{tx_d} - \theta_1 + \phi_{\delta 1}), \quad (2.13)$$

com $\phi_{\delta 1} = -\phi_{rx_1} + \phi_{tx_1}$.

O sensor S_0 recebe esse sinal (Equação (2.13)) e, após conversão para banda base, sincroniza seu PLL₂ gerando

$$r_0(t) = \cos(\omega_{bb}t + \phi_{tx_d} - \theta_1 + \phi_{\delta 1} - \theta_{ss} - \phi_{rx_0}). \quad (2.14)$$

Durante o *time-slot* 3, ambos os sensores transmitem, simultaneamente, os sinais de seus PLL₂ como

$$tx_0(t) = \cos(\omega t + \phi_{tx_d} - \theta_1 + \phi_{\delta 1} - \theta_{ss} + \phi_{\delta 0}), \quad (2.15)$$

e

$$tx_1(t) = \cos(\omega t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0} - \theta_{ss} + \phi_{\delta 1}). \quad (2.16)$$

O sinal $tx_0(t)$ (Equação (2.15)) é recebido pelo interrogador como

$$rx_{d0}(t) = \cos(\omega t + \phi_{tx_d} - \theta_1 + \phi_{\delta 1} - \theta_{ss} + \phi_{\delta 0} - \theta_0), \quad (2.17)$$

e o sinal $tx_1(t)$ (Equação (2.16)) é recebido pelo interrogador como

$$rx_{d1}(t) = \cos(\omega t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0} - \theta_{ss} + \phi_{\delta 1} - \theta_1). \quad (2.18)$$

Os sinais $rx_{d0}(t)$ (Equação (2.17)) e $rx_{d1}(t)$ (Equação (2.18)) representam o resultado da propagação pelos percursos ($D \rightarrow S_0 \rightarrow S_1 \rightarrow D$) e ($D \rightarrow S_1 \rightarrow S_0 \rightarrow D$).

O sinal efetivamente recebido pelo interrogador durante o intervalo TS_3 é $rx_d(t) = rx_{d0}(t) + rx_{d1}(t)$ resultando em

$$rx_d(t) = 2\cos(\omega t + \phi_{tx_d} - \theta_0 + \phi_{\delta 0} - \theta_{ss} + \phi_{\delta 1} - \theta_1), \quad (2.19)$$

que é uma combinação coerente de $rx_{d0}(t)$ e $rx_{d1}(t)$.

Em uma rede composta de $N > 2$ sensores, apenas os sensores S_0 e S_{N-1} propagam uma extensão periódica do sinal piloto. Os elementos intermediários, $S_1 - S_{N-2}$, retransmitem os sinais recebidos de seus pares em ambas as direções e combinam as fases desses sinais com a fase do sinal recebido diretamente do interrogador para gerar o sinal de transmissão.

2.8 Conclusão

Os esquemas de transmissão cooperativa em redes multi-hop (*Ad-Hoc*) prometem o uso eficiente de energia, aumento da área coberta pela rede e, mais importante, podem aumentar o tempo de vida dessas redes pela redução da potência durante a transmissão e redução do consumo de energia das baterias. A implementação desses esquemas, entretanto, requer a existência de uma camada física compatível, que permita o controle de parâmetros como frequência e fase. Considerando a existência dessa camada física, diferentes métodos podem ser utilizados para implementar transmissão de forma coerente melhorando a relação sinal/ruído e permitindo comunicações mais eficientes em termos de distância ou consumo de energia.

Neste capítulo foram apresentados os fundamentos dos principais métodos de *beam-forming* distribuídos encontrados na literatura. Os métodos existentes foram divididos e classificados em duas categorias: malha fechada, em que o interrogador participa ativamente do processo de sincronização e malha aberta em que o interrogador fornece apenas o sinal piloto de treinamento. Em seguida, descreveu-se um método representativo de cada categoria, com especial atenção para o método baseado em malha aberta utilizado como referência para comparação com o método proposto nesta tese e apresentado no próximo capítulo.

Capítulo 3

Método Proposto

O método proposto nessa tese é baseado na ideia de responder a um interrogador com o conjugado complexo de um sinal de sincronismo, *beacon*, enviado por esse interrogador. O modelo simplificado de rede considerado e o princípio conceitual do método são apresentados nas duas próximas seções e o método proposto, propriamente dito, é discutido nas seções subsequentes.

A abordagem aqui proposta, é inspirada no funcionamento das antenas refletoras ou arranjos retrodiretivos de antenas [65]-[68], que respondem direcionando um feixe em direção ao interrogador sem a necessidade de calcular o ângulo de chegada.

Nos sistemas de antenas refletoras, o direcionamento do feixe de retorno é conseguido pela transmissão, por cada elemento transmissor, de um sinal cuja fase de saída é obtida pela inversão, em cada elemento receptor, do ângulo de fase de chegada do sinal piloto transmitido pelo interrogador. A transmissão simultânea, por cada elemento transmissor, do sinal correspondente com a fase invertida, garante o cancelamento do efeito do canal entre o interrogador e cada elemento do arranjo de antenas refletoras, resultando no interrogador, em uma recepção coerente dos sinais transmitidos por cada elemento do arranjo.

3.1 Modelo simplificado do sistema

O sistema considerado está ilustrado na Figura 2.3 (por conveniência, aqui reproduzida como Figura 3.1) e é composto de um interrogador (D) com potência suficiente para alcançar os N sensores S_i ($i = 0, \dots, N - 1$). Os transceptores do interrogador e dos sensores possuem osciladores locais (LO) independentes com fases iniciais iguais a ϕ_D e ϕ_i , respectivamente.

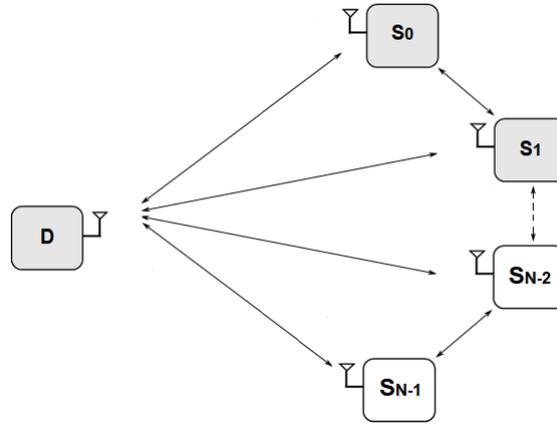


Figura 3.1: Diagrama de rede com 1 interrogador e N sensores.

Como o interesse principal nessa tese é o alinhamento de fase, assume-se que os elementos da rede se encontram perfeitamente sincronizados em frequência pelo uso compartilhado de um sinal de referência de 10 MHz, escolhido de forma *ad hoc*. Em um cenário mais realista, diversos métodos para sincronismo global de frequência estão disponíveis, como discutidos por D. C. Rife et al, em [57], e por Suleiman et al, em [58], e mesmo sistemas de distribuição de *clock* sem fio podem ser considerados como aqueles apresentados por Abari et al, em [59], S. Mghabghab et al, em [60] e por R. L. Schmid et al, em [61].

Considerando um canal sem seletividade em frequência e quasi-estático (sem alterações durante o tempo de um quadro de RF), modelado como um atraso de tempo, e sinais de banda estreita, tem-se que o efeito do canal sobre os sinais será equivalente a um deslocamento de fase que pode ser representado como θ_i , para o deslocamento de fase entre o interrogador e o sensor S_i e θ_{ij} para o deslocamento de fase entre os sensores S_i e S_j e considera-se que $\theta_{ij} = \theta_{ji}$.

O sistema opera sob o paradigma pergunta/resposta, em dois estágios, com o interrogador (D) transmitindo um sinal interrogando os sensores (S_0 e S_1) que respondem de forma sincronizada. Uma sequência de ciclos de pergunta/resposta representando um experimento completo é chamada de *campanha*. Um ciclo completo de pergunta-resposta é composto de dois estágios (conforme representado na Figura 3.2):

- Estágio 1 - Sincronismo – a estação de destino, interrogador (D), transmite, em modo de difusão, um sinal piloto (*beacon*) para sincronismo de frequência e fase, seguida de símbolos BPSK (*Binary Phase Shift Keying*) para endereçamento dos sensores.
- Estágio 2 - *Beamforming* – após se sincronizarem os sensores (S_0 e S_1) respondem de forma simultânea, modulando uma portadora derivada do sinal de *beacon* com a fase invertida. Nas próximas seções o método proposto será descrito em maiores detalhes.

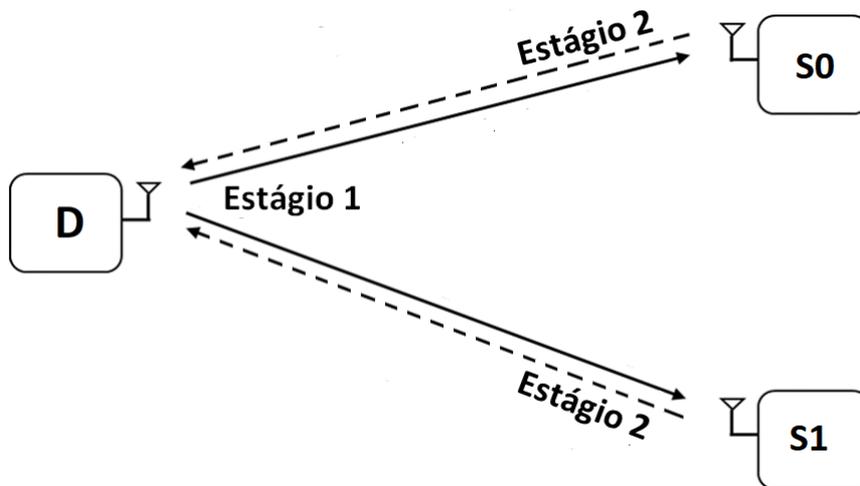


Figura 3.2: Modelo simplificado do sistema.

3.2 Método Proposto - Princípio básico

O método para alinhamento de fase baseado no cancelamento da fase do canal, ilustrado na Figura 3.3, pode ser descrito como a seguir:

Representando o sinal piloto, enviado pelo interrogador, como um fasor (tx) com fase inicial igual a zero (estágio I), e que gira no sentido anti-horário à medida que o sinal se desloca no espaço, tem-se que os sinais percebidos pelos nós sensores podem ser representados por fasores (r_0 e r_1) com fases iniciais que estão relacionadas com o caminho percorrido pelo sinal piloto entre o interrogador e cada sensor (estágio II). Os conjugados complexos desses fasores (estágio III) resultam em fasores (s_0 e s_1) que, se continuarem girando, simultaneamente, na mesma direção, esses fasores se alinharão (rx_0 e rx_1) no ponto inicial do círculo unitário (estágio IV). Se esses fasores conjugados representarem os sinais de transmissão dos nós sensores, tem-se que esses sinais de retorno serão somados em fase no interrogador (estágio V), resultando em sinal com amplitude igual à soma das amplitudes dos sinais individuais (rx).

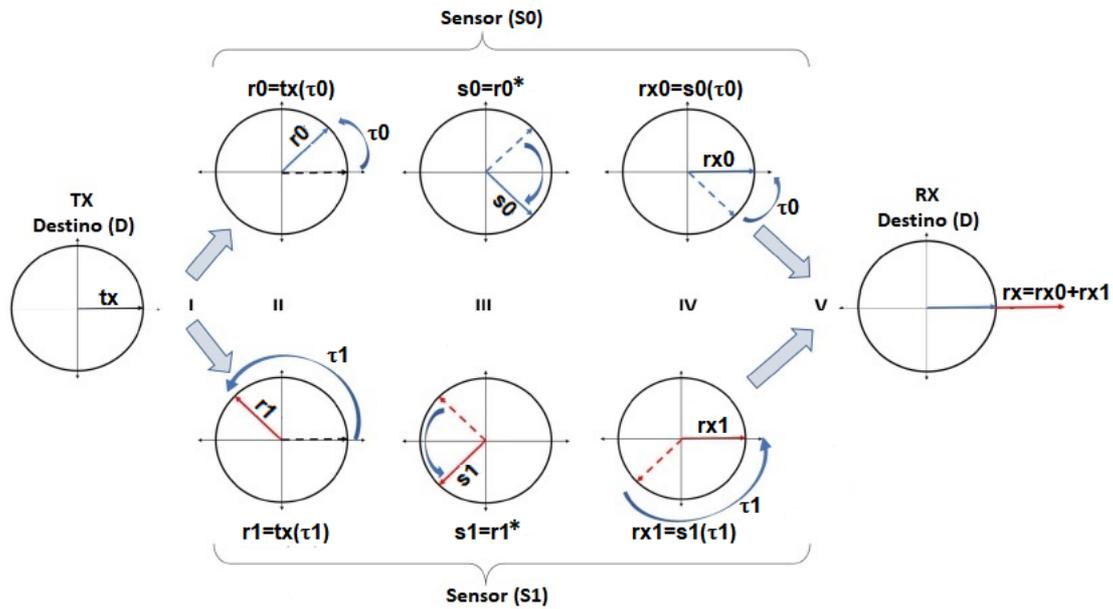


Figura 3.3: Processo de alinhamento de fase pela inversão da fase de chegada do sinal piloto.

Esse processo pode ser descrito formalmente como:

Considere um sinal piloto (*beacon*), $\cos(2\pi f_b t)$, em banda base, convertido para frequência de RF e transmitido pelo interrogador (D) como

$$Tx(t) = \cos(2\pi f_{RF} t + \phi_D), \quad (3.1)$$

com $f_{RF} = f_{lo} + f_b$, f_b representando a frequência do sinal de *beacon*, f_{lo} representando a frequência do oscilador local (OL) com $f_{lo} \gg f_b$.

Como a fase inicial do sinal do interrogador, ϕ_D , é a mesma para todos os sensores e como a amplitude do sinal não é crítica para o sincronismo dos PLLs implementados nos sensores, para efeito de simplificação, ϕ_D e a amplitude do sinal piloto podem ser considerados iguais a zero e um, respectivamente.

Devido ao atraso do percurso, os sensores S_i ($i = 0, 1$) recebem o sinal de *beacon* como

$$r_i(t) = Tx(t - \tau_i) = \cos(2\pi f_{RF}t - \theta_i), \quad (3.2)$$

com τ_i igual ao atraso devido ao percurso, $\theta_i = 2\pi d_i/\lambda \bmod(2\pi)$ igual ao deslocamento de fase equivalente e d_i a distância entre o interrogador (D) e o sensor S_i .

Durante o estágio de sincronização, fazendo uso de um PLL sintonizado em f_b , a partir do sinal $r_i(t)$, o sensor S_i recupera o sinal em banda base,

$$rbb_i(t) = \cos(2\pi f_b t - \theta_i), \quad (3.3)$$

assumindo, idealmente, que a fase inicial do oscilador local de recepção, ϕ , seja igual a zero.

Durante o estágio de *beamforming*, o sensor estima o ângulo de chegada θ_i e gera o sinal banda base para transmissão com a fase invertida

$$tbb_i(t) = \cos(2\pi f_b t + \hat{\theta}_i), \quad (3.4)$$

que após a conversão para frequência de RF, é transmitido como

$$s_i(t) = \cos(2\pi f_{RF}t + \hat{\theta}_i) = \cos(2\pi f_{RF}t + \hat{\theta}_i). \quad (3.5)$$

O receptor do interrogador (D) receberá o sinal do sensor S_0 como $Rx_0(t)$ e receberá o sinal do sensor S_1 como $Rx_1(t)$.

Considerando que nem o interrogador (D) nem os sensores sofreram deslocamentos físicos

durante o processo, os sinais transmitidos pelos sensores percorrerão o mesmo caminho de volta, sofrendo o mesmo atraso de tempo no retorno, resultando em

$$Rx_0(t) = \cos(2\pi f_{RF}t + \hat{\theta}_0 - \theta_0), \quad (3.6)$$

e

$$Rx_1(t) = \cos(2\pi f_{RF}t + \hat{\theta}_1 - \theta_1). \quad (3.7)$$

Portanto, considerando que a distância entre D e S_i é muito maior que a distância entre S_0 e S_1 , tem-se que $Rx_0(t) \simeq Rx_1(t)$.

Logo, o sinal efetivamente recebido pelo interrogador (D), será uma combinação coerente de $Rx_0(t)$ e $Rx_1(t)$ igual a

$$Rx(t) = Rx_0(t) + Rx_1(t) \simeq 2\cos(2\pi f_{RF}t), \quad (3.8)$$

que tem o dobro da amplitude do sinal representado pela Equação (3.1).

Dessa forma a operação de inversão de fase, combinada com o caminho reverso percorrido pelo sinal, termina por anular o efeito da diferença de fase entre os sensores, e seus sinais se combinam de forma coerente no destino, de forma semelhante ao que acontece nos arranjos de antenas refletoras.

Entretanto, nos sistemas de antenas refletoras, que são sistemas centralizados, o ângulo de fase é determinado a partir da relação conhecida entre as antenas e o referenciamento a um oscilador local comum a todas as antenas. Por outro lado, RSSF são sistemas distribuídos e seus elementos possuem osciladores locais distintos e independentes e tanto suas fases iniciais quanto a relação entre essas fases são desconhecidas, não existindo um sinal comum de referência, nem relação conhecida entre os elementos (antenas) que permita deduzir o ângulo de chegada em cada elemento.

Do ponto de vista do processamento em banda base em cada elemento, o ângulo de

fase de chegada do sinal do interrogador, é composto de duas partes: um deslocamento de fase causado pelo percurso e um deslocamento de fase causado pelo processo de conversão de frequência (*downconverter*), que depende diretamente da fase inicial do oscilador local. A simples inversão desse ângulo de fase, criaria, em cada elemento da rede, um sinal de transmissão com fase diferente e não relacionada impedindo a coerência de fase no ponto de recepção, conforme ilustrado na Figura 3.4. Apenas a parte da fase correspondente ao percurso, θ , deve ser invertida, e para isso é necessário separa-la da componente de fase correspondente ao oscilador local, ϕ .

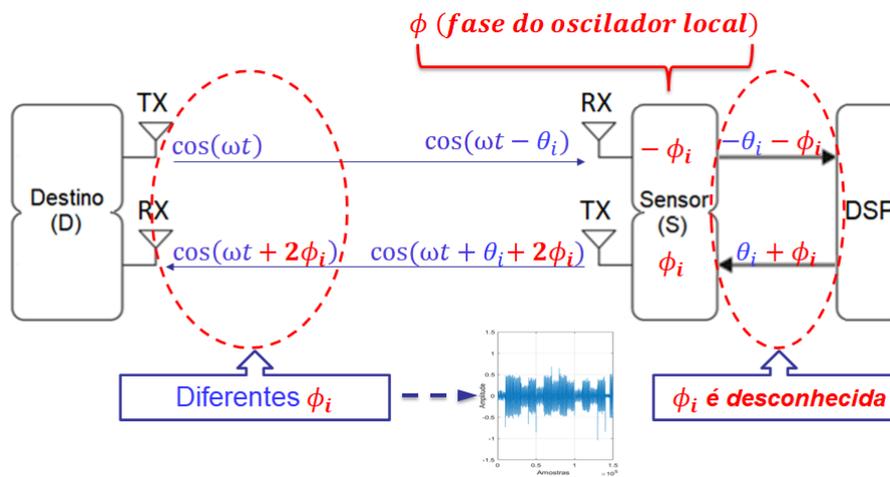


Figura 3.4: Combinação não-coerente do sinal no interrogador devido à inversão, indesejada, da fase inicial e desconhecida do oscilador local.

3.3 Método Proposto (Inversão Remota de Fase - IRF)

Considerando a rede ilustrada na Figura 3.1, e considerando um cenário mais realista, cada transceptor utiliza dois osciladores locais (OL) independentes com fases iniciais para transmissão e recepção iguais a ϕ_{Tx_D} , ϕ_{Tx_i} e ϕ_{Rx_D} , ϕ_{Rx_i} , respectivamente. Ignorando os termos ϕ_{Tx_D} e ϕ_{Rx_D} por serem constantes e não contribuírem para diferenças relativas de fase e, fazendo $\omega = 2\pi f_{RF}$ para simplificar a notação, o sinal piloto transmitido pelo interrogador, $Tx_d(t) = \cos(\omega t)$, é recebido pelos sensores como $Rx_i(t) = \cos(\omega t - \theta_i - \phi_{Rx_i})$. A transmissão de volta, simultânea, desse sinal pelo conjunto de sensores S_i será percebida pelo interrogador como

3.3.1 Inversão Remota de Fase (IRF)

Sequencialmente, utilizando um protocolo Mestre-Escravo, cada sensor S_i ($i = 1..N - 1$) replica o sinal $Rx_i(t)$ para S_0 que é recebido como

$$Rx_{0i}(t) = \cos(\omega t - \theta_i - \phi_{Rx_i} + \phi_{Tx_i} - \theta_{i0} - \phi_{Rx_0}). \quad (3.11)$$

A fase desse sinal é invertida por S_0 gerando

$$Rx_{0i_inv}(t) = \cos(\omega t + \theta_i + \phi_{Rx_i} - \phi_{Tx_i} + \theta_{i0} + \phi_{Rx_0} + \phi_{Tx_0}), \quad (3.12)$$

que é transmitido de volta por S_0 para S_i que o recebe como

$$Rx_{i_inv}(t) = \cos(\omega t + \theta_i - \phi_{Tx_i} + \phi_{Rx_0} + \phi_{Tx_0}). \quad (3.13)$$

Fazendo $Tx_i(t) = Rx_{i_inv}(t)$ ($i = 1..N - 1$), tem-se

$$Tx_i(t) = \cos(\omega t + \theta_i - \phi_{Tx_i} + \phi_{Rx_0} + \phi_{Tx_0}) \quad (3.14)$$

para ($i = 1..N - 1$).

E fazendo $Tx_0(t)$ igual a $Rx_0(t)$ com fase invertida tem-se

$$Tx_0(t) = \cos(\omega t + \theta_0 + \phi_{Rx_0}). \quad (3.15)$$

Quando todos os sensores, S_i ($i = 0, \dots, N - 1$), transmitem $Tx_i(t)$ simultaneamente, o sinal recebido pelo interrogador devido a transmissão de S_i ($i = 1..N - 1$) é expresso por

$$Rx_{d_i}(t) = \sum_{i=1}^{N-1} \cos(\omega t + \phi_{Rx_0} + \phi_{Tx_0}), \quad (3.16)$$

e o sinal recebido devido a transmissão de S_0 é dado por

$$Rx_{d_0}(t) = \cos(\omega t + \phi_{Rx_0} + \phi_{Tx_0}), \quad (3.17)$$

portanto

$$Rx_d(t) = \sum_{i=0}^{N-1} \cos(\omega t + \gamma_i), \quad (3.18)$$

com $\gamma_i = \phi_{Rx_0} + \phi_{Tx_0} = \text{constante}$, maximizando a Equação (3.10).

3.3.2 Minimização de Latência

O sinal a ser transmitido, $Tx_i(t) = \cos(\omega t + \theta_i - \phi_{Tx_i} + \phi_{Rx_0} + \phi_{Tx_0})$, é equivalente a uma inversão local da fase do sinal piloto recebido $Rx_i(t) = \cos(\omega t - \theta_i - \phi_{Rx_i})$ com uma rotação adicional de fase $\Psi_i = -\phi_{Rx_i} - \phi_{Tx_i} + \phi_{Rx_0} + \phi_{Tx_0}$, que se mantém constante durante toda a campanha e que pode ser determinada pela inversão remota da fase de um sinal $Tx_{cal_i}(t) = \cos(\omega t)$ gerado localmente durante um estágio prévio de calibração.

Com base nessa observação, estabeleceu-se um protocolo para minimização de latência.

O protocolo é dividido em dois estágios: **Calibração**, com $2N - 2$ *time-slots* por campanha, ilustrado esquematicamente na Figura 3.6 e **Operação**, com 2 *time-slots* por ciclo, ilustrado esquematicamente na Figura 3.7.

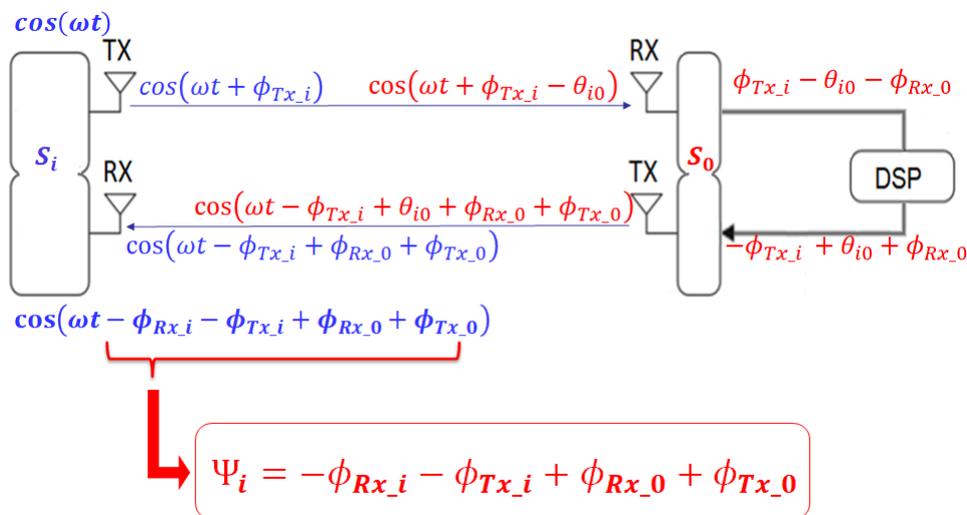


Figura 3.6: Estágio de calibração.

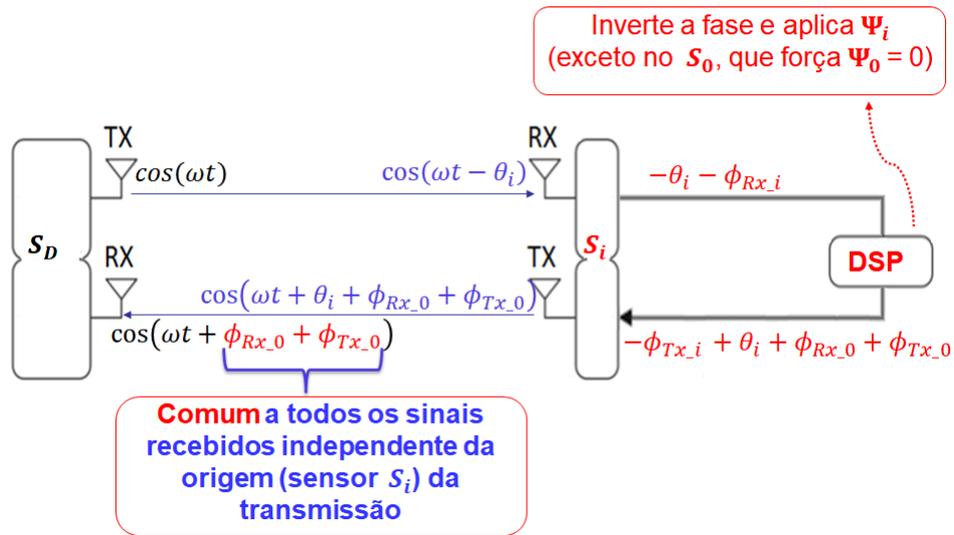


Figura 3.7: Estágio de operação.

O protocolo pode ser descrito como:

Estágio de calibração ($k = 1..N$):

TS_{2k-1} : S_k transmite $Tx_{cal_k}(t) = \cos(\omega t)$ para S_0 que recebe

$$Rx_{cal_k}(t) = \cos(\omega t + \phi_{Tx_k} - \theta_{k0} - \phi_{Rx_0}). \quad (3.19)$$

Esse sinal tem sua fase invertida por S_0 gerando

$$Rx_{cal_k_inv}(t) = \cos(\omega t - \phi_{Tx_k} + \theta_{k0} + \phi_{Rx_0}). \quad (3.20)$$

TS_{2k} : S_0 transmite $Rx_{cal_k_inv}(t)$ para S_k , que recebe

$$Rx_{cal_0k_inv}(t) = \cos(\omega t - \phi_{Rx_k} - \phi_{Tx_k} + \phi_{Rx_0} + \phi_{Tx_0}). \quad (3.21)$$

A diferença de fase entre $Rx_{cal_0k_inv}(t)$ e $Tx_{cal_k}(t)$ é calculada por S_k gerando

$$\Psi_k = -\phi_{Rx_k} - \phi_{Tx_k} + \phi_{Rx_0} + \phi_{Tx_0}, \quad (3.22)$$

que é armazenada para uso posterior durante o estágio de operação.

Estágio de operação:

TS_0 : O interrogador transmite um sinal piloto que é recebido por cada sensor S_i como

$$Rx_i(t) = \cos(\omega t - \theta_i - \phi_{Rx_i}). \quad (3.23)$$

Esse sinal tem sua fase invertida por S_i , gerando

$$Rx_{i_inv}(t) = \cos(\omega t + \theta_i + \phi_{Rx_i}). \quad (3.24)$$

TS_1 : Os sensores S_i fazem $Tx_i(t) = Rx_{i_inv}(t)e^{j(\Psi_i)}$ (com $\Psi_0 = 0$) e transmitem simultaneamente

$$Tx_0(t) = \cos(\omega t + \theta_0 + \phi_{Rx_0})$$

$$Tx_i(t) = \cos(\omega t + \theta_i - \phi_{Tx_i} + \phi_{Rx_0} + \phi_{Tx_0})$$

para $(i = 1..N - 1)$, cujo resultado no interrogador é representado pela Equação (3.18).

Para uma mesma campanha, o primeiro ciclo do método IRF ocupa o mesmo número de *time-slots* que o método RT, mas a partir do segundo ciclo o número de *time-slots* por ciclo se mantém constante = 2, independente do número de sensores, conforme ilustrado na Figura 3.8, para um sistema hipotético composto de 3 sensores e 1 interrogador.

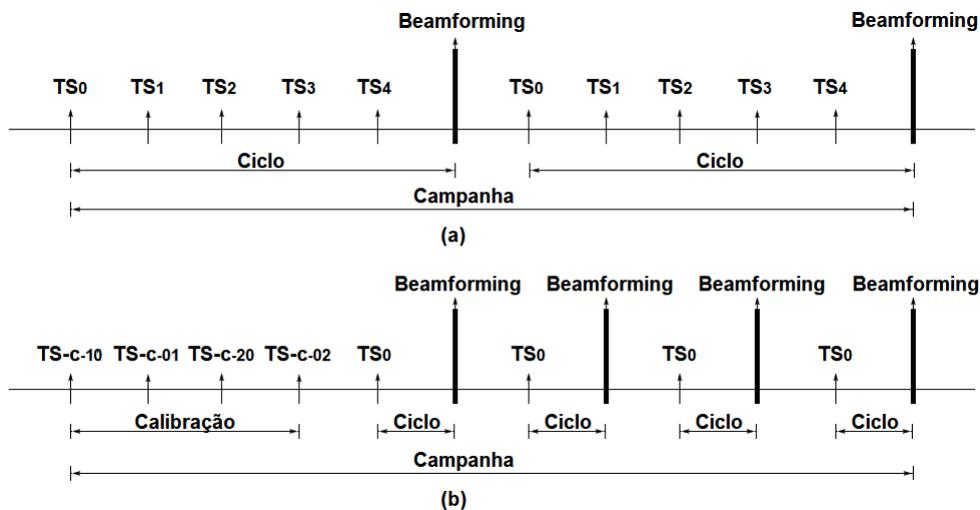


Figura 3.8: Relação temporal: (a) *Round-trip*, (b) *Inversão Remota de Fase*.

Para uma mesma taxa de *bits*, o método IRF, após o estágio de calibração, necessita

de um menor número de intervalos ativos de tempo para transmitir um mesmo volume de dados quando comparado com o método RT, o que pode se traduzir em menor consumo de energia.

3.4 Conclusão

Neste capítulo o princípio do método proposto foi apresentado e, após o detalhamento de sua forma geral, foi descrita uma otimização que reduz significativamente sua latência após um estágio de calibração.

Capítulo 4

Testes e Resultados

Neste capítulo, inicialmente são apresentados resultados obtidos por meio de simulação, validando o princípio básico do método de cancelamento do efeito do canal pela inversão da fase do sinal de *beacon* enviado pelo interrogador e recebido pelos sensores. Em seguida, após a descrição da configuração para testes, são apresentados os resultados obtidos experimentalmente.

Os testes simulados e experimentais foram realizados utilizando um modelo simplificado de rede, composto de 1 interrogador e 2 sensores, representada pela parte hachureada da Figura 4.1 (por conveniência, aqui reproduzida da Figura 2.3).

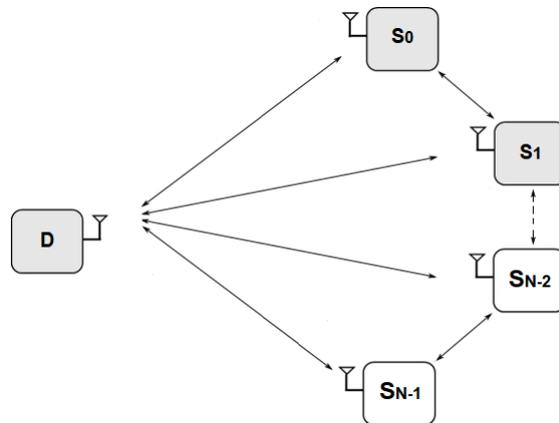


Figura 4.1: Diagrama de rede com 1 interrogador e N sensores.

4.1 Resultados obtidos por meio de Simulação

Inicialmente foi simulado, em Matlab, o mecanismo baseado no princípio básico de sincronismo por inversão de fase e, em seguida, foi simulada uma aplicação utilizando esse princípio.

4.1.1 Princípio básico

A simulação foi implementada considerando uma rede composta de três elementos (um interrogador e dois sensores) e um canal ideal, modelado como um atraso de propagação em função da distância. Considerando sinais de banda estreita, o atraso de propagação é equivalente a um deslocamento de fase e a estimativa dessa fase foi assumida como perfeita. Na Figura 4.2 está destacado o deslocamento de fase experimentado pelo sinal durante o trânsito entre o interrogador e os sensores. O traço azul representa o sinal de *beacon* transmitido pelo interrogador (D) e os traços vermelho e amarelo representam o sinal de *beacon* como recebido pelo sensores S_0 e S_1 , respectivamente. Como os sensores se encontram posicionados a diferentes distâncias do interrogador, cada um recebe o sinal com um deslocamento de fase diferente, θ_0 para o sensor S_0 e θ_1 para o sensor S_1 .

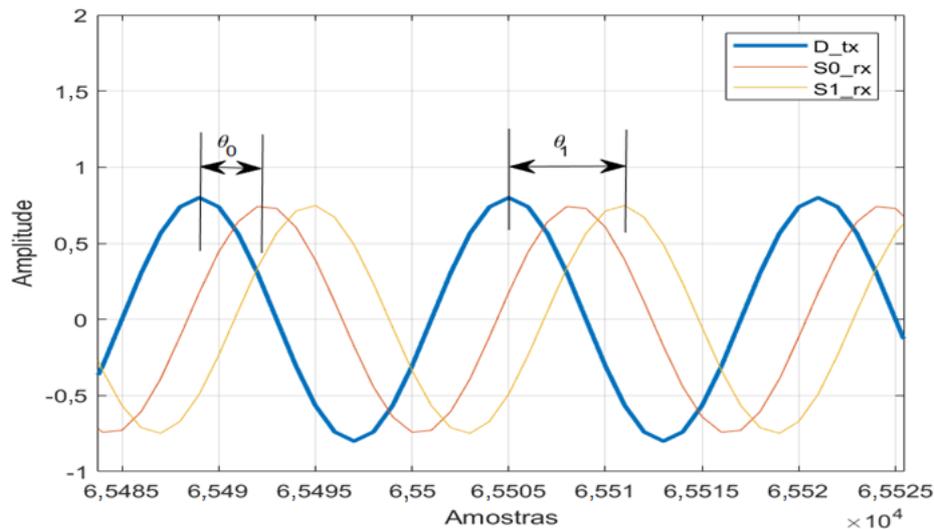


Figura 4.2: Diferença de fase entre o sinal de *beacon* enviado pelo interrogador e o sinal recebido por cada sensor.

Nas Figuras 4.3(a) e 4.3(b) estão ilustradas, de forma isolada para cada sensor, as

sequências de operações do algoritmo. Após a estimativa do deslocamento de fase, $\hat{\theta}_i$, a correção de fase $2\hat{\theta}_i$ é aplicada ao sinal recebido (traço vermelho) resultando no sinal com fase invertida (tracejado azul). Esse sinal é transmitido de volta para o interrogador e, como ele é submetido ao mesmo atraso de propagação, chega em fase com o sinal original no interrogador (tracejado preto, levemente fora de escala para destaque visual).

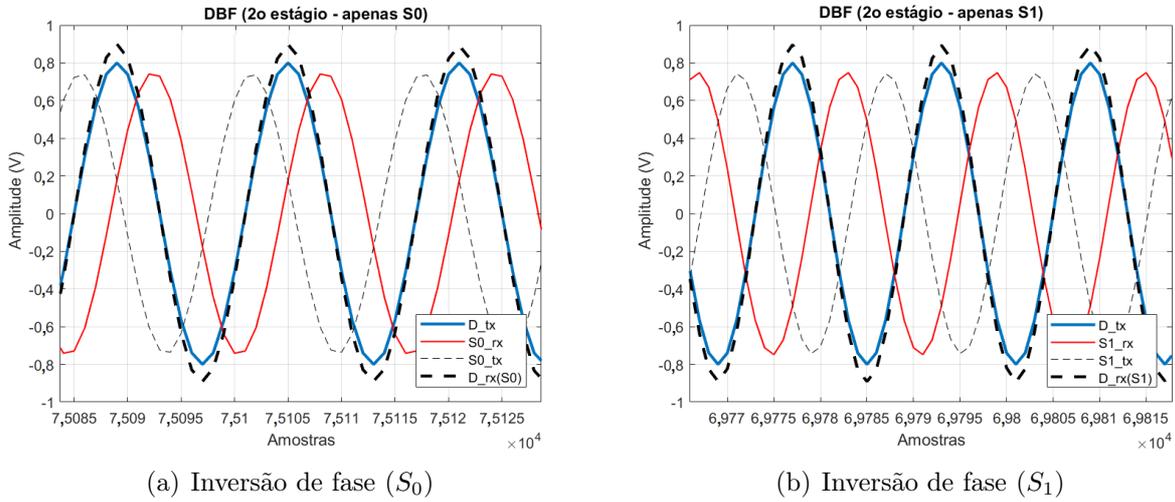


Figura 4.3: Simulação do processo de alinhamento de sinal por inversão de fase.

E, na Figura 4.4, está ilustrada a aplicação desse processo aos dois sensores simultaneamente, e a recepção combinada dos dois sinais no destino (interrogador), resultando em um sinal com o dobro da amplitude de cada sinal individual (tracejado preto, em escala).

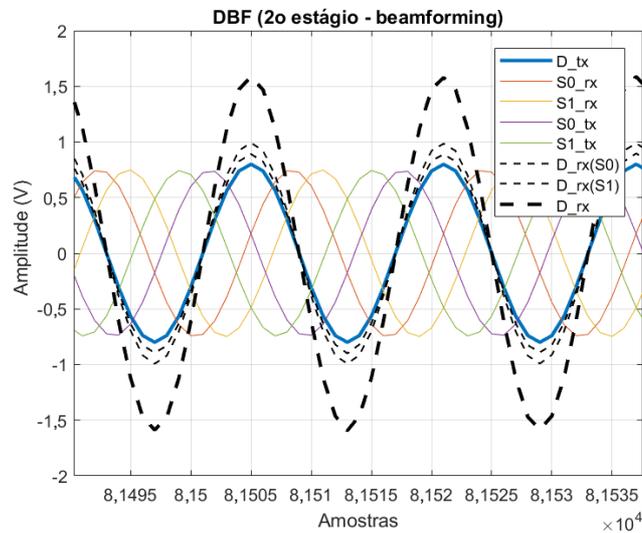


Figura 4.4: Relação de fase entre os sinais durante recepção combinada no destino.

4.1.2 Aplicação

Posteriormente, uma simulação foi elaborada considerando que o sinal de *beacon* passa por um *upconverter* no transmissor do destino, passa por um *downconverter* nos receptores dos sensores e, após a inversão de fase, passa por um *upconverter* nos transmissores dos sensores. Considerou-se canais AWGN (*Additive White Gaussian Noise*) assimétricos destino/sensores e sensores/destino. O canal no sentido destino/sensores foi simulado com relação sinal-ruído de 40 dB e o canal no sentido sensores/destino foi simulado com uma relação sinal-ruído variando de -4 a +4 dB em passos de 1 dB.

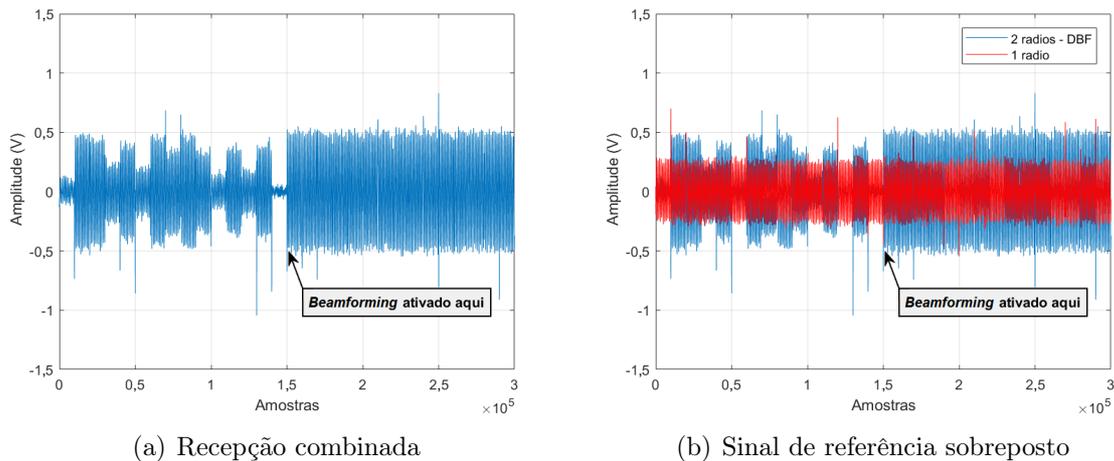
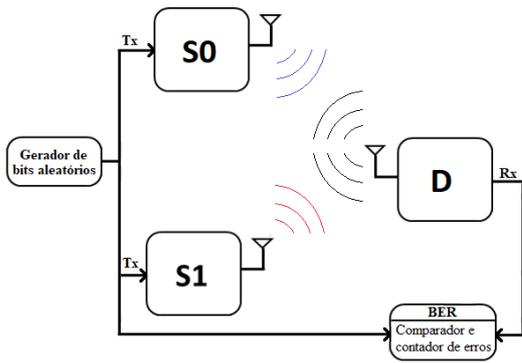


Figura 4.5: Resultado da simulação de *Beamforming* Distribuído via Inversão de Fase

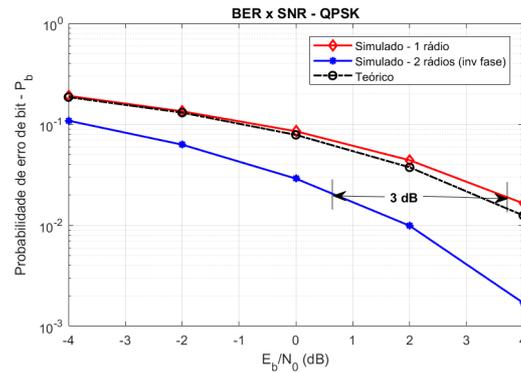
Na Figura 4.5(a) está apresentada a recepção combinada dos sinais de dois transmissores e, para referência, na Figura 4.5(b) o sinal de apenas um rádio foi sobreposto ao sinal combinado. Com o objetivo de destacar o efeito do mecanismo de *beamforming*, a Figura 4.5 foi gerada a partir de um procedimento em que após receberem o sinal de *beacon*, os dois rádios sensores iniciam a transmissão de forma independente e sem sincronismo até que tenham transmitido 145.000 amostras. É possível notar que os sinais se combinam, aleatoriamente, hora de forma construtiva (em torno da amostra de número 80.000, por exemplo) e hora de forma destrutiva (pouco antes da amostra 100.000). Após a amostra de número 145.000 os rádios sincronizam seus PLLs e ativam seus respectivos módulos de correção de fase. Pode-se notar o momento em que o *beamforming* é ativado (em torno da amostra de número 150.000) quando os rádios sincronizam seus PLLs, ativam seus respectivos mó-

dulos de correção de fase e passam a transmitir no modo de *beamforming*. A partir desse momento, os sinais se combinam de forma coerente, resultando em um sinal com o dobro da amplitude de cada sinal quando considerados isoladamente, como era de se esperar, de acordo com a Equação (3.8).

Essa observação é consistente com o ganho de 3 dB observado nos testes de taxa de erro de bit, BER (*Bit Error Rate*), obtidos por simulação com o esquema mostrado na Figura 4.6(a) e apresentados na Figura 4.6(b), quando se aplica uma modulação BPSK ao sinal de *beacon* de retorno com fase corrigida.



(a) Esquema utilizado



(b) BER em função de SNR

Figura 4.6: Avaliação da taxa de erro de bits em função da Relação Sinal/Ruído

4.2 Resultados Experimentais

O arranjo experimental utilizado é composto de três dispositivos de *hardware USRP-N2x0*, conectados a um computador via gigabit ethernet com protocolo TCP/IP. Esse arranjo está representado pelo diagrama de blocos apresentado na Figura 4.7. Cada *USRP* utiliza uma interface de RF *RFX900* sintonizada na frequência de 925 MHz. A taxa de amostragem do sinal em banda-base foi programada para 1 MS/s.

As distâncias entre o interrogador *D* e os sensores S_0 e S_1 são 3,4 m e 3,2 m, respectivamente e a distância entre os sensores é de 60 cm.

Em todos os testes, os protocolos foram ligeiramente modificados para permitir a captura do sinal de cada sensor individualmente, após a correção de fase mas antes do *beamforming*, para visualização das formas de onda no destino (interrogador).

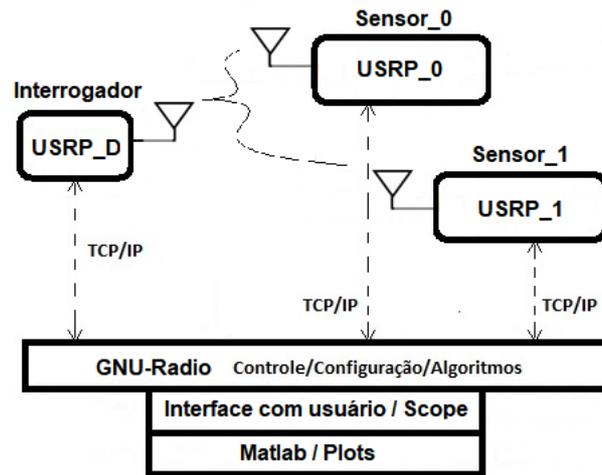


Figura 4.7: Diagrama em blocos do arranjo experimental.

4.2.1 Inversão de Fase (IRF) com fase conhecida

O primeiro experimento teve por objetivo validar experimentalmente o princípio básico de inversão de fase, similar ao experimento simulado da Seção 4.1.1. Para isso foi utilizado o esquema apresentado na Figura 4.8 (descrito no Apêndice A), em que a fase a ser invertida, θ_i , é constante, conhecida e foi deduzida diretamente da medição das distâncias entre os sensores e o interrogador segundo a expressão $\theta_i = 2\pi d_i/\lambda \pmod{2\pi}$.

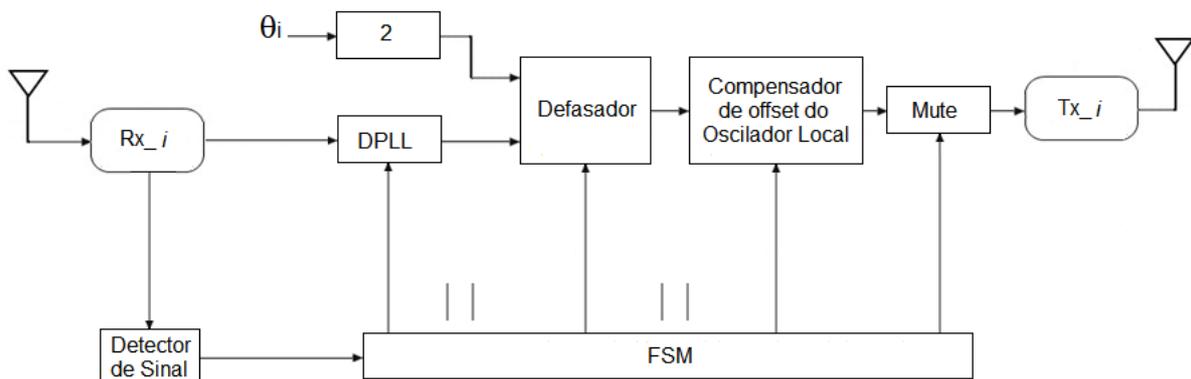


Figura 4.8: Diagrama de blocos dos receptores para o experimento IRF com fase conhecida.

Ao detectar o sinal de *beacon* enviado pelo interrogador, a *FSM* do sensor ativa o PLL que se sincroniza com o sinal de *beacon*, e, assumindo que ele tem a forma $\cos(\omega t + \phi_{tx_D} - \theta_i - \phi_{rx_i})$, ativa o bloco *defasador* para gerar o sinal $\cos(\omega t + \phi_{tx_D} + \theta_i - \phi_{rx_i})$, efetivamente invertendo θ_i . Em seguida, para compensar o efeito das fases iniciais dos osciladores locais,

o *compensador de offset* do OL acrescenta uma rotação adicional de fase de $\phi_{rx_i} - \phi_{tx_i}$, determinada previamente durante o estágio de calibração. O sinal $\cos(\omega t + \phi_{tx_D} + \theta_i)$, é então transmitido pelo sensor de volta para o interrogador.

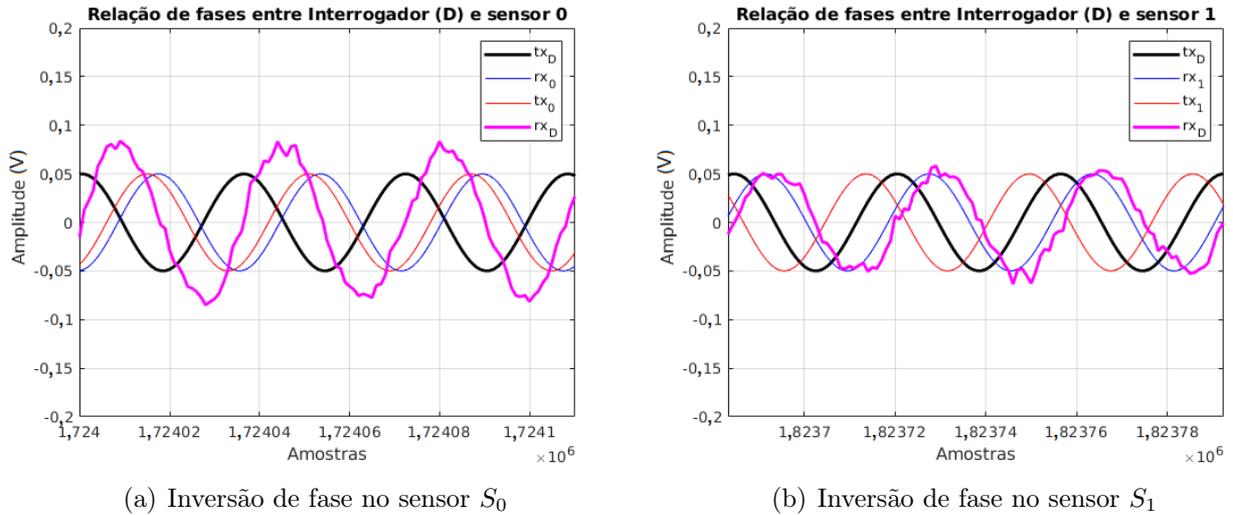


Figura 4.9: Alinhamento de fase.

Esse processo está ilustrado nas Figuras 4.9(a) e 4.9(b) para os sensores S_0 e S_1 , respectivamente. Diferentemente do exemplo simulado, não é possível perceber visualmente a relação correta de fase entre os sinais pois, nas figuras, eles se apresentam sob efeito das fases iniciais dos osciladores locais. O traço preto representa o sinal de *beacon* (referência). O traço azul representa o sinal recebido após o *downconverter* e o traço vermelho representa o sinal após a rotação de fase e antes do *upconverter*. O traço de cor magenta representa o sinal visto pelo receptor do interrogador (destino).

Comparando-se os sinais de ambos os sensores com a mesma referência, é possível perceber que os sinais recebidos (traços azuis) tem fases diferentes mas que, ao final do processo, chegam com fases iguais no receptor de destino (traços magentas).

Os sinais mostrados nas Figuras 4.9(a) e 4.9(b), foram gerados ativando cada sensor individualmente, primeiro S_0 e depois S_1 . Quando ambos os sensores são ativados simultaneamente, o resultado da combinação coerente no destino pode ser visto na Figura 4.10, com a amplitude do sinal resultante (traço magenta) aproximadamente igual à soma das amplitudes dos sinais quando considerados individualmente.

Na Figura 4.11 está representada uma campanha composta de dois ciclos do protocolo

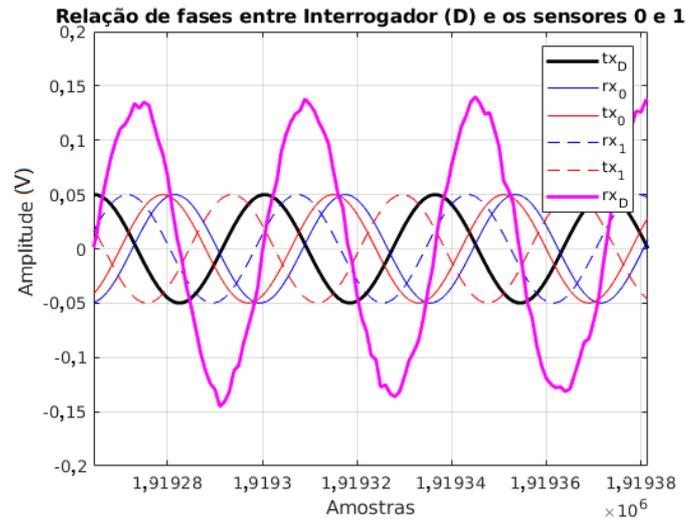


Figura 4.10: Relação de fase entre os sinais durante recepção combinada no destino.

em que se vê o nível do sinal na forma de Indicador de intensidade do sinal recebido (RSSI), calculado como o módulo da magnitude do sinal, ao quadrado. As *etiquetas* S_0 e S_1 indicam a ativação individual de cada sensor para efeito de comparação com o resultado combinado (*etiqueta* $S_0 + S_1$).

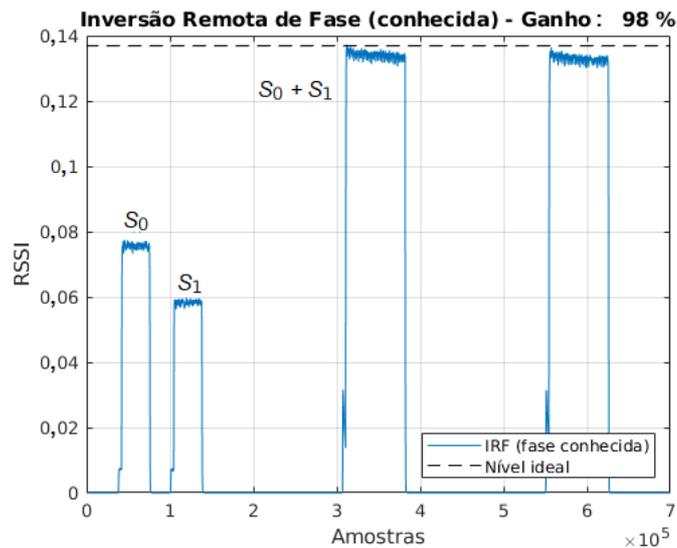


Figura 4.11: Resultado do método IRF quando a fase é conhecida.

Esse método resultou, de forma consistente, em um ganho de *beamforming*, definido como $RSSI_{medido} / (RSSI_{S_0} + RSSI_{S_1})$, igual a 98%.

4.2.2 Inversão remota de fase (IRF)

O segundo experimento teve por objetivo validar experimentalmente o método de alinhamento de portadora via inversão de fase em sua forma geral (fase desconhecida), conforme descrito no Capítulo 3. O esquema utilizado está mostrado na Figura 4.12, cuja implementação está descrita no Apêndice A.

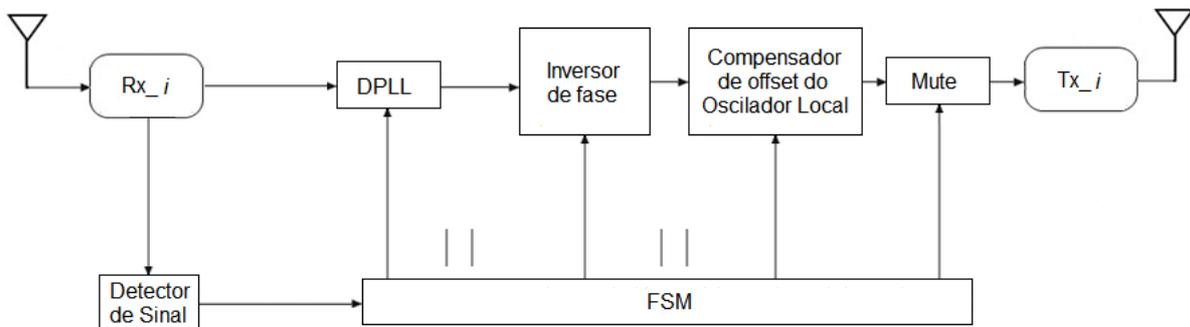
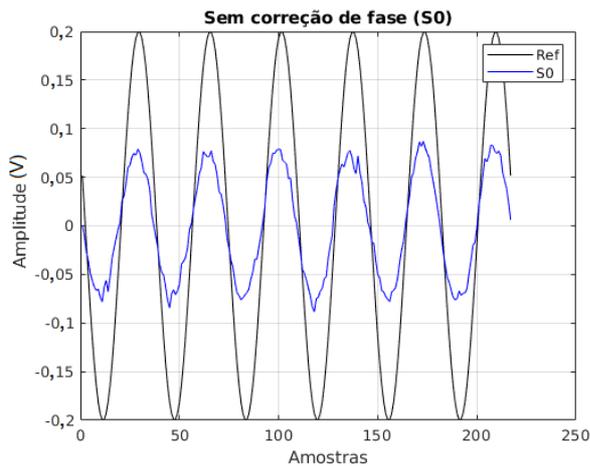


Figura 4.12: Diagrama de blocos dos receptores para o experimento IRF.

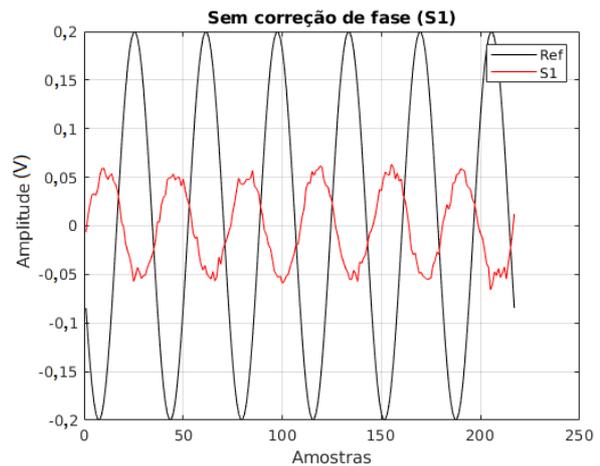
Inicialmente o experimento foi realizado com o protocolo IRF desativado com o objetivo de observar a combinação não-coerente dos sinais no receptor do interrogador. Nesse modo, os blocos *inversor de fase* e *compensador de offset* do OL ficaram desativados e, ao detectar o sinal de *beacon*, a *FSM* do sensor ativa o PLL que se sincroniza ao sinal de *beacon*. Em seguida, esse mesmo sinal é transmitido de volta para o interrogador sem nenhuma correção de fase. O resultado desse procedimento pode ser visto nas Figuras 4.13 e 4.14.

Nas Figuras 4.13(a) e 4.13(b), podem ser vistas as recepções dos sinais dos sensores S_0 e S_1 , respectivamente. O traço preto representa um sinal de referência para comparação de fase. O traço azul representa o sinal do sensor S_0 e o traço vermelho representa o sinal do sensor S_1 . O alinhamento de fase do sinal do sensor S_0 com a referência não tem nenhuma relevância, pois é apenas uma coincidência desse experimento em particular. O fato relevante a ser observado é que os sinais se apresentam com fase diferente quando se compara com o sinal de referência, como esperado. Essa diferença de fase entre os sinais, vista na Figura 4.14(a), tem como consequência uma recepção não-coerente do sinal resultante.

A ativação simultânea dos dois sensores, resulta, no receptor, em um sinal não-coerente como se pode ver na Figura 4.14(b), com a amplitude do sinal resultante, traço magenta, neste caso em particular, menor que a amplitude do sinal do sensor S_0 .

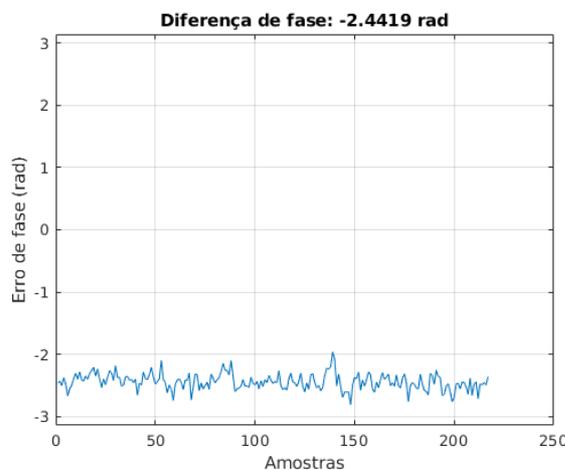


(a) Sinal do sensor S_0 sem correção de fase

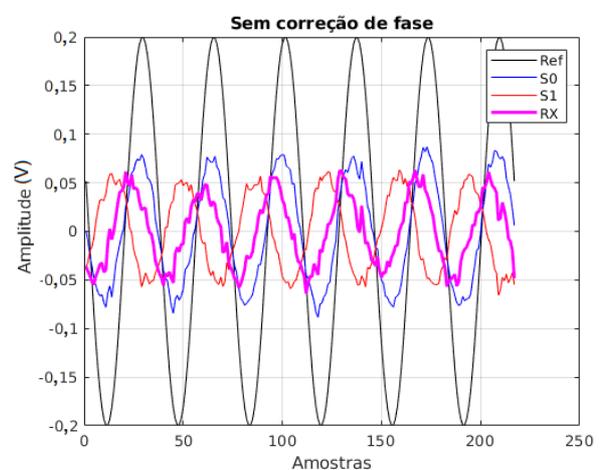


(b) Sinal do sensor S_1 sem correção de fase

Figura 4.13: Relação de fase entre os sinais dos sensores no receptor (sem correção de fase).



(a) Diferença absoluta de fase (Erro)

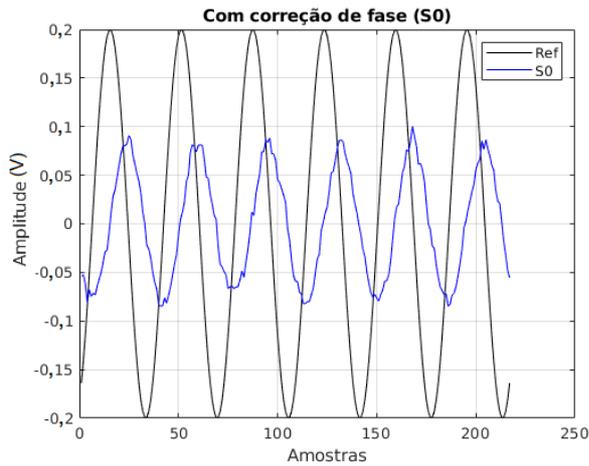


(b) Recepção sem correção de fase

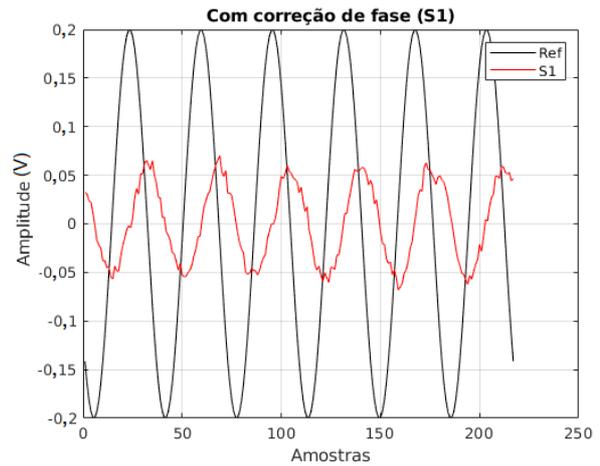
Figura 4.14: Recepção combinada sem correção de fase.

Quando o protocolo IRF foi ativado, obteve-se as corretas correções de fase necessárias para uma recepção coerente no interrogador, como se pode ver nas Figuras 4.15 e 4.16.

Pode-se notar que a fase do sinal S_0 , Figura 4.15(a), é aproximadamente igual à do sensor S_1 (Figura 4.15(b)). Uma vez que a diferença de fase entre os sinais, vista na Figura 4.16(b), é pequena, a combinação dos dois sinais resulta em um sinal com amplitude maior que as amplitudes de cada um dos sinais quando considerados individualmente, como se vê na Figura 4.16(a), com a amplitude do sinal resultante, traço magenta, aproximadamente igual à soma das amplitudes de S_0 e S_1 .

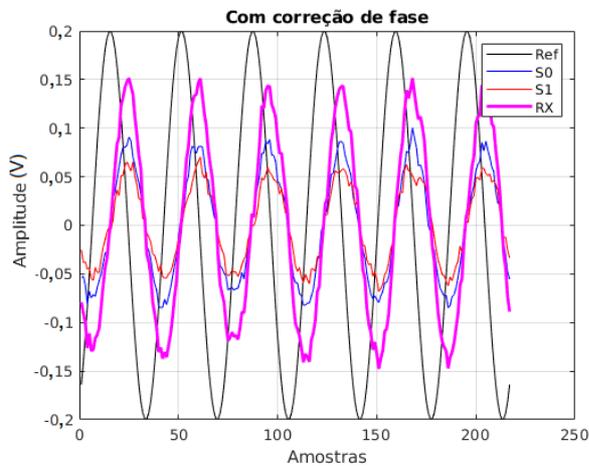


(a) Sinal do sensor S_0 após correção de fase

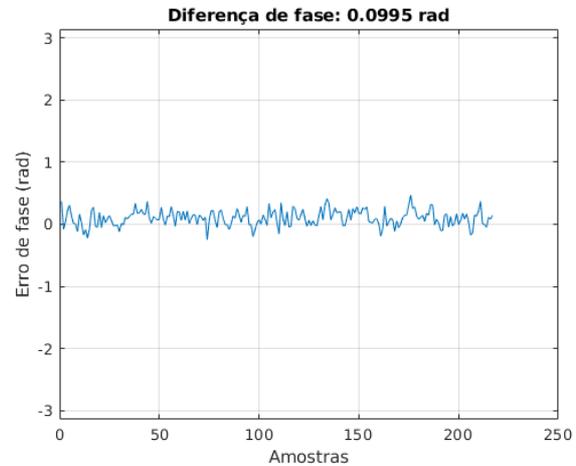


(b) Sinal do sensor S_1 após correção de fase

Figura 4.15: Relação de fase entre os sinais dos sensores no receptor (após a correção de fase).



(a) Recepção (com correção de fase)



(b) Diferença absoluta de fase (Erro)

Figura 4.16: Recepção combinada com correção de fase.

Na Figura 4.17 está representada uma campanha composta de cinco ciclos de repetição, com o ganho de *beamforming* se mantendo consistentemente acima de 97%.

Este desempenho se manteve quando se realizou uma campanha composta de 100 ciclos, como se pode ver nas Figuras 4.18(a) e 4.18(b), que retratam a evolução da diferença de fase entre os sensores e do ganho de *beamforming*, respectivamente.

Neste experimento, os rádios foram reinicializados a cada 10 ciclos. A cada reinicialização, os osciladores locais são sintonizados e apresentam novas fases iniciais aleatórias, o que

garante que os resultados estejam livres de vieses ou tendências.

As diferenças de fase, coletadas a cada ciclo, se situaram entre $4,2623 \times 10^{-5}$ e $0,1252$ radianos, com uma média de $0,036587$ radianos e variância de $0,00074492$. O ganho de *beamforming* variou de $98,0189$ a $99,7972$ %, com valor médio de $98,7387$ % e variância de $0,074514$.

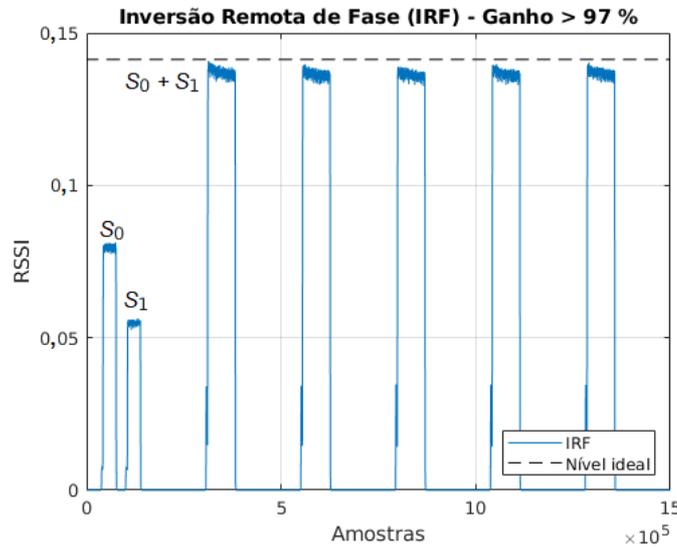
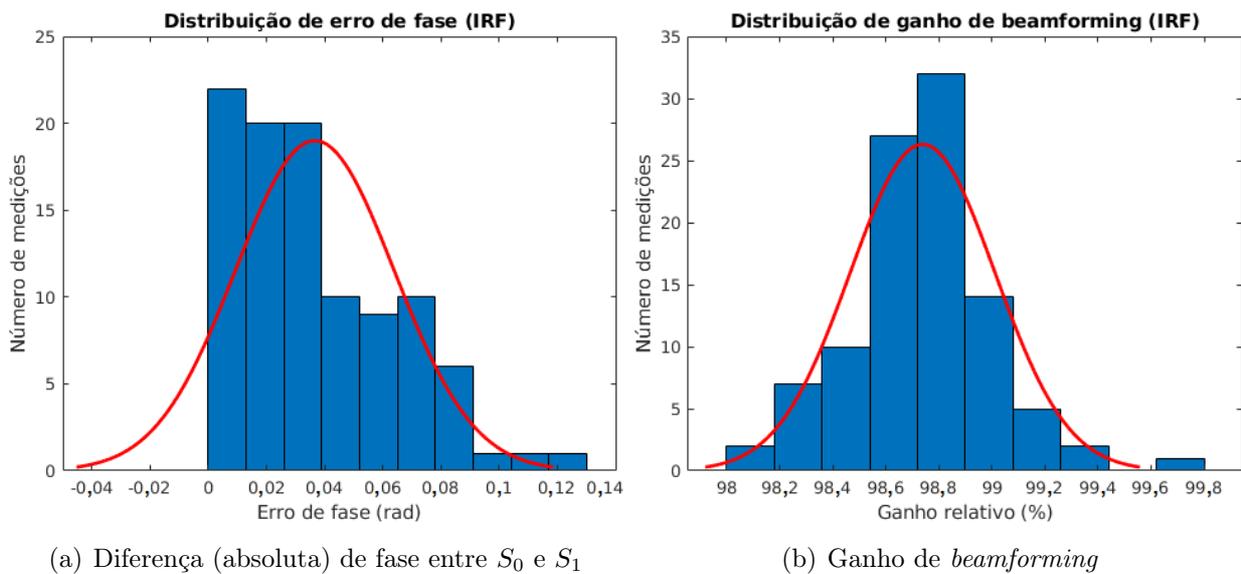


Figura 4.17: Experimento IRF - Intensidade do sinal recebido pelo interrogador.



(a) Diferença (absoluta) de fase entre S_0 e S_1

(b) Ganho de *beamforming*

Figura 4.18: Distribuição de ganho e erro de fase (campanha composta de 100 ciclos).

4.2.3 Time-slotted Round Trip (RT)

Os testes com o método *Time-slotted Round Trip* (RT), descrito no Capítulo 2 e usado como referência, foram efetuados utilizando o receptor representado pelo diagrama de blocos visto na Figura 4.19 e descrito no Apêndice A.

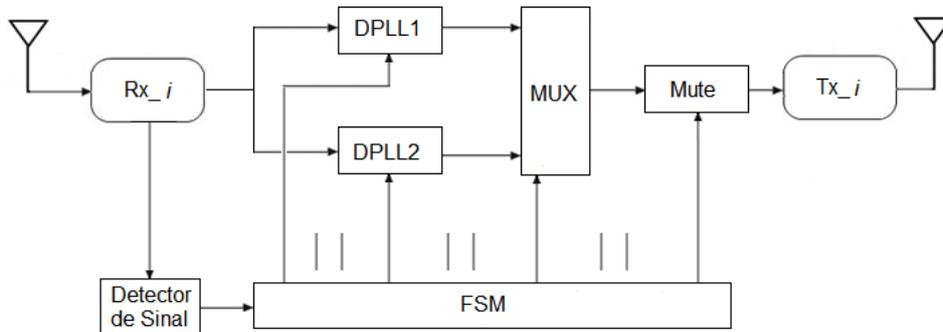


Figura 4.19: Diagrama de blocos dos receptores para o método RT.

O resultado de uma campanha com cinco ciclos é apresentado na Figura 4.20, comprovando a correta implementação do protocolo que apresentou ganho de *beamforming* maior que 97% durante toda a campanha.

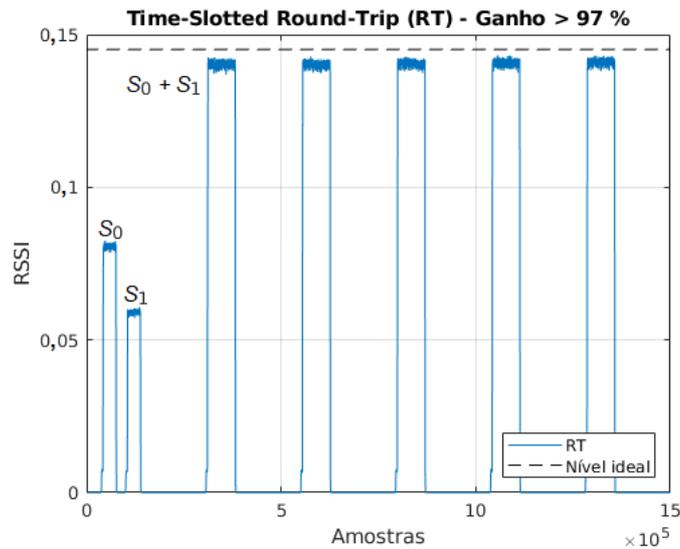


Figura 4.20: Experimento RT - Intensidade do sinal recebido pelo interrogador.

Nas Figuras 4.21(a) e 4.21(b) estão representadas as distribuições de diferença de fase e ganho de *beamforming*, respectivamente, em uma campanha maior, composta de 100 ciclos.

Neste experimento, observou-se que o erro de fase variou de 0,010982 a 0,32237 radianos, com valor médio de 0,12238 radianos, e o ganho de *beamforming* variou de 96,2966 a 99,7618%, com valor médio de 98,9182%.

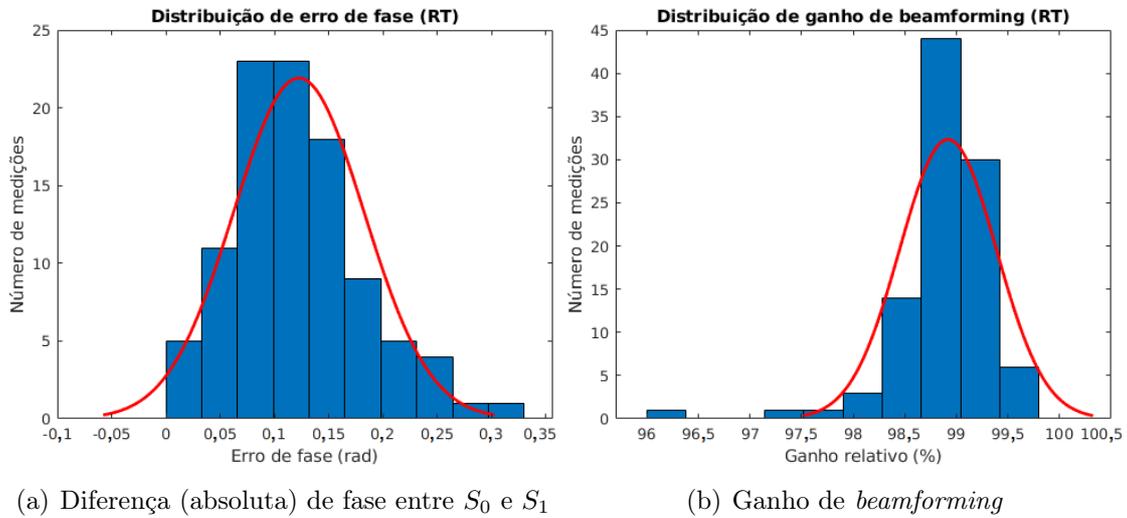


Figura 4.21: Distribuição de ganho e erro de fase (campanha com 100 ciclos).

4.2.4 One Bit Feedback (OBFB)

O experimento com o método *One Bit Feedback*, descrito no Capítulo 2, foi baseado em [38, 56, 64] e o arranjo utilizado está representado pelo diagrama de blocos visto na Figura 4.22, em que a variação de fase se dá em passos que variam, aleatoriamente, entre 0 e $\pi/8$.

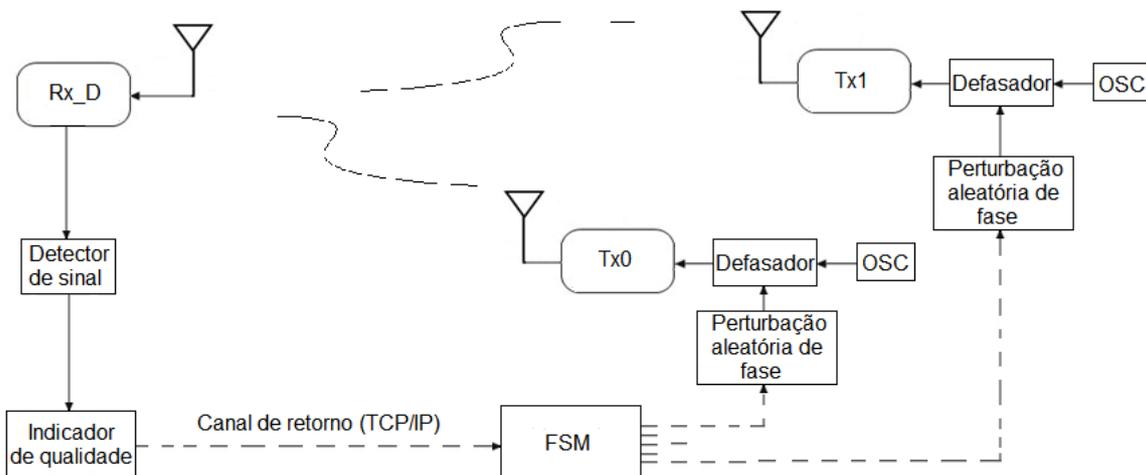


Figura 4.22: Diagrama em blocos da implementação do experimento *One Bit Feedback*.

Neste protocolo não faz muito sentido se falar em campanha com um certo número de ciclos, uma vez que o protocolo, por definição, opera repetitivamente até que o nível de sinal detectado pelo interrogador alcance um valor arbitrário, determinado pelo usuário, como pode ser visto na Figura 4.23.

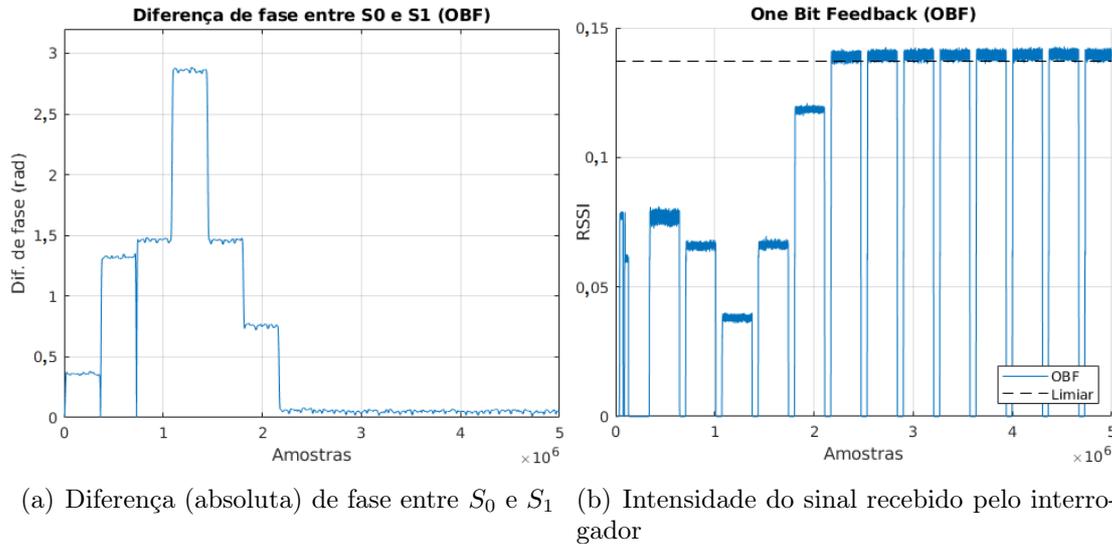


Figura 4.23: Variação de fase e intensidade do sinal no experimento *One Bit Feedback*.

É possível notar, claramente, uma correlação entre a diferença de fase entre os sinais S_0 e S_1 , vista na Figura 4.23(a) e evolução do nível de sinal, vista na Figura 4.23(b), com destaque para o instante logo após a amostra de número 1.000.000 quando a diferença de fase se encontra próximo de π , caracterizando um interferência fortemente destrutiva, e o nível do sinal com valor menor que o nível de qualquer dos sinais quando considerados individualmente. Outro momento interessante se situa logo após a amostra de número 2.000.000 quando a diferença de fase se aproxima de zero, quando, então o nível de sinal alcança o limiar desejado.

4.2.5 Comparações

IRF x OBF

Uma comparação entre esses dois métodos (IRF e OBF) tem validade apenas funcional, se alcançam ou não seus objetivos, uma vez que por suas próprias naturezas não permitem uma comparação direta de desempenho. Enquanto o método IRF procura atingir o menor

erro de fase e maior ganho possíveis, o método OBF busca uma convergência de fase que leve o nível de sinal até um limiar pré-determinado.

Na Figura 4.24 pode-se observar que ambos os métodos alcançam seus objetivos e fica evidente a grande latência do método OBF quando comparado ao método IRF.

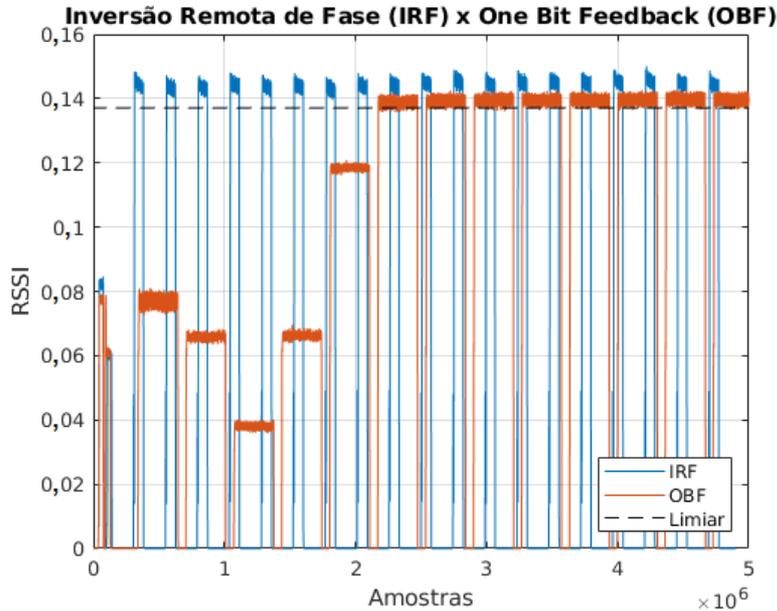


Figura 4.24: Comparação entre os resultados dos métodos IRF e OBF.

IRF x RT

Os dois métodos (RT e IRF) apresentam resultados semelhantes, quando se compara as amplitudes das Equações (2.19) e (3.18), como se pode ver na Figura 4.25 com ambos os métodos apresentando ganhos de *beamforming* de 97% do ganho ideal. Entretanto o método RT apresenta maior oscilação no ganho ao longo do tempo, como visto na Tabela 4.1, em que a variância de ganho no método RT é três vezes maior que a variância apresentada pelo ganho no método IRF, o que pode ser parcialmente explicado, entre outros fatores, pelo maior número de estimativas de fase inerente ao protocolo, com maior erro acumulado. A variância da diferença de fase no método RT, vista na Tabela 4.2 é cinco vezes maior que a variância de diferença de fase do método IRF.

Interrompendo a repetição de ciclos e mantendo constante o estado de *beamforming*, a manutenção da curva do IRF próximo ao ideal, visto na Figura 4.26, sugere que o método

IRF suporta um intervalo de tempo maior entre sequências de treinamento, o que pode representar menor consumo de energia para um mesmo volume de dados a ser transmitido.

Tabela 4.1: Índices de ganho de *Beamforming* do método RT e do método proposto, IRF

Método	Mínimo	Máximo	Média	Variância
RT	96,2966	99,7618	98,0292	0,21967
IRF	98,0189	99,7972	98,7387	0,074514

Tabela 4.2: Índices de erro de fase no método (RT) e no método proposto, (IRF)

Método	Mínimo	Máximo	Média	Variância
RT	0,010982	0,32237	0,12238	0,0036074
IRF	4,2623e-05	0,1252	0,036587	0,00074492

Outra comparação preliminar entre os métodos pode ser vista na Tabela 4.3 com destaque para uma menor latência apresentada pelo método IRF bem como para uma menor complexidade para implementação.

A redução de latência do método IRF quando comparada à latência do método RT, deve-se à divisão da campanha em dois estágios: calibração (determinação e inversão de

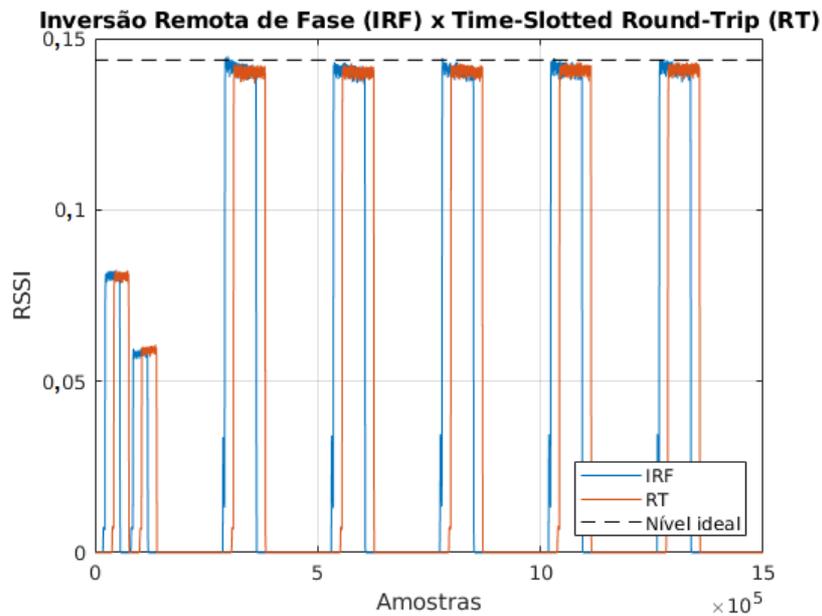


Figura 4.25: Comparação entre os resultados dos métodos RT e IRF.

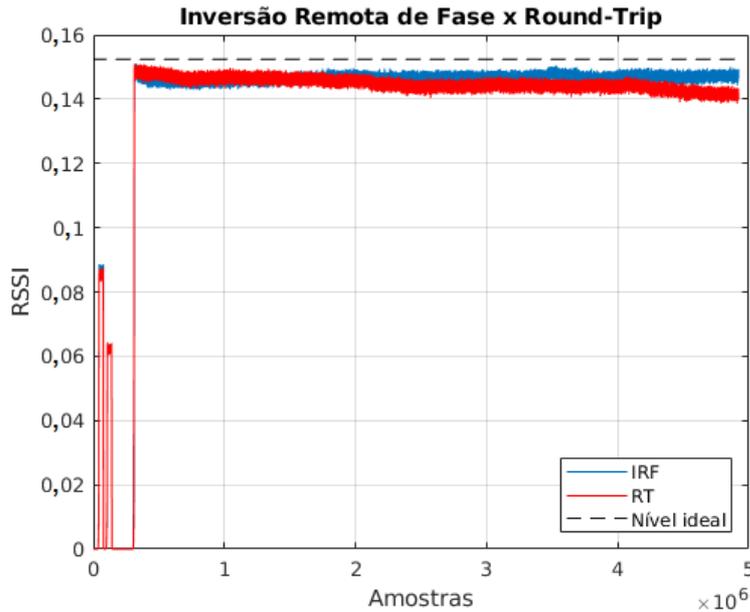


Figura 4.26: Variação de ganho nos métodos IRF e RT.

Tabela 4.3: Comparação entre os métodos One-Bit Feedback (OBF), Time-slotted Round-trip (RT) e o método proposto, Inversão Remota de Fase (IRF)

Método	<i>Time Slots</i>	<i>Latência</i>	<i>Complexidade</i>
OBF	Não determinístico	Não determinístico	Baixa
RT	$2 N^*$	Cresce com N^*	Referência
IRF	2^{**}	Baixa, constante ^{**}	Mais baixa que a referência

*Número de elementos sensores. **A partir do segundo ciclo.

fase do sinal) e operação (transmissão no modo de *beamforming*), concentrando mecanismo de calibração no primeiro ciclo da campanha e mantendo apenas o processo de *beamforming* durante os ciclos subsequentes.

A conclusão de que o método IRF apresenta menor complexidade de implementação é intuitiva, uma vez que, por exemplo, o transceptor utilizado no método RT necessita de dois PLLs enquanto que o transceptor do método IRF utiliza apenas um PLL, que é usado tanto para a calibração quanto para transmissão no modo de *beamforming*.

4.3 Conclusão

Os resultados obtidos experimentalmente foram coerentes com os resultados obtidos por simulação e validaram tanto o método proposto quanto sua implementação. Foi possível demonstrar, por comparação, que o método proposto apresenta melhor estabilidade que o método de referência e apresenta, potencialmente, menor consumo de energia devido ao menor número de *time-slots* necessários para completar um ciclo de comunicação.

Capítulo 5

Conclusão

Nessa tese foi apresentado um método para alinhamento de fase de portadora visando a implementação de *beamforming* distribuído em redes de sensores sem fio (RSSF). Um arcabouço de software baseado em GNU Radio foi construído para validação experimental desse método e foi utilizado na implementação de uma prova de conceito de três métodos de *beamforming* distribuído: um sistema em malha fechada, denominado *One-bit Feedback Synchronization*, um sistema em malha aberta denominado *Time-Slotted Round-Trip Synchronization*, utilizado como referência, e o método proposto, também em malha aberta, denominado *Inversão Remota de Fase*.

A princípio, o método proposto (IRF), pode ser considerado como uma variação do método *Round Trip* (RT), ao mesmo tempo em que guarda alguma semelhança com o método *Two-Way* (TW). Entretanto os três métodos diferem nos seguintes aspectos:

- O método RT alcança coerência de fase considerando o trânsito do sinal por percursos com atrasos equivalentes, em que um sinal de treinamento enviado pelo interrogador é circulado entre os sensores;
- No método TW, de forma similar ao método RT, a circulação de um sinal de treinamento entre os elementos sensores é utilizada no processo de sincronização. Entretanto esse sinal de treinamento é gerado pelos sensores e não pelo interrogador;
- No método IRF o deslocamento de fase do sinal devido ao percurso é cancelado pela

inversão dessa fase, após o cancelamento dos deslocamentos iniciais dos osciladores locais, efetuado localmente entre os sensores.

Quando comparado aos métodos RT e TW, que alcançam o alinhamento de fase pela propagação de estimativas de fase de sensor a sensor, com consequente acúmulo de erros de estimação, o método IRF utiliza apenas um sensor específico para estimativa e inversão das fases iniciais dos osciladores locais, o que reduz os erros acumulados de estimação de fase. Essa redução no erro acumulado de fase tem como consequência um ganho de *beamforming* mais estável, permitindo que o processo de estimação de fase fique concentrado no início dos ciclos de uma determinada campanha. Como os valores de fase estimados, no método IRF, se mantém inalterados por mais tempo que no método RT, o tempo entre treinamentos e estimação de fase é maior, o que reduz significativamente a latência a partir do segundo ciclo de pergunta/resposta.

A validação do princípio básico do método se deu por simulação e, posteriormente, o sistema foi validado experimentalmente sobre a plataforma de hardware USRP utilizando o arcabouço de software construído especificamente para essa tese. Os resultados demonstram que:

- o método proposto apresenta melhor desempenho em termos de estabilidade e de ganho,
- bem como apresenta menor latência quando comparado com o método de referência e
- a hipótese levantada no Capítulo 1, de que os métodos de *beamforming* utilizados em arranjos refletores, baseados na inversão da fase de chegada do sinal, podem ser adaptados para uma abordagem descentralizada, foi validada uma vez que no método proposto, o alinhamento de fase dos sinais de resposta dos sensores é alcançado pela inversão da fase de chegada do sinal piloto, mesmo que de forma indireta, garantindo recepção coerente pelo interrogador.

A validação do método *Inversão Remota de Fase* (IRF) e as implementações de demonstração dos outros dois métodos mencionados, em uma plataforma de hardware combinada

com o arcabouço apresentado, comprovaram a utilidade e versatilidade dos novos módulos funcionais, inexistentes na biblioteca original da plataforma de software utilizada e representam uma contribuição adicional à comunidade GNU Radio, com destaque para os módulos *phase shifter* (deslocador de fase) e *fsm* (controlador sequencial).

Em GNU-Radio, um deslocamento de fase pode ser provocado em um sinal pela multiplicação complexa desse sinal por uma constante complexa. Entretanto esse mecanismo não pode ser utilizado se o deslocamento de fase desejado for dinâmico. Essa deficiência é eliminada pela utilização do módulo *phase shifter*, que pode variar o deslocamento de fase de amostra a amostra.

Por ser uma ferramenta originalmente voltada para o processamento de fluxos contínuos de amostras (*streams*), GNU-Radio não oferece mecanismos intuitivos para o processamento sequencial das amostras, o que dificulta o desenvolvimento de aplicações dependentes da relação entre eventos temporais, como por exemplo aplicações *half-duplex*. O módulo *fsm* facilita a implementação desse tipo de aplicação uma vez que permite, de forma seletiva, a ativação e desativação dos diversos módulos que compõem a aplicação. Diferentes algoritmos podem ser testados sem modificação no *grafo* da aplicação, bastando apenas alterar o arquivo texto que representa sua tabela de transição de estados.

Em adição ao método IRF, nessa tese, é apresentado um estudo preliminar sobre um método alternativo denominado *Inversão de Fase baseada em Duplo Tom* (IFDT), cuja principal motivação é a redução da latência para apenas dois *time-slots*, independente do número de sensores. No Apêndice B é apresentada a descrição desse método com resultados preliminares obtidos por meio de simulação.

5.1 Considerações e Sugestão para estudos futuros

Nessa tese considerou-se um ambiente estático, em que nem os sensores nem o interrogador se movem. Na prática seria interessante levar-se em consideração um contexto com mobilidade entre os elementos, o que significa estender a interpretação dos canais de comunicação e considerar aspectos relevantes como efeito doppler e desvanecimento causado por

multipercurso.

Considerando que mesmo com maior estabilidade no processo de estimação de fase no método IRF, o desalinhamento de fase é inevitável devido a variações nos osciladores locais, é desejável o desenvolvimento de um protocolo que, periodicamente, analise a qualidade do ganho de *beamforming* e oriente os sensores para repetirem o processo de calibração.

A escolha do nó sensor responsável pela inversão de fase dos sinais (mestre) também foi deixada em aberto nessa tese, oferecendo oportunidade para o desenvolvimento de um método ou protocolo, provavelmente adaptativo e periódico para a determinação desse elemento.

A investigação de mecanismos para o sincronismo distribuído de frequência sem a necessidade de hardware adicional oferece oportunidade para novos desdobramentos, pois a complexidade dos sistemas de RF poderia ser reduzida e permitiria a implementação completa de um sistema distribuído.

O uso do termo *sensores* para os elementos que cooperativamente participam do processo de *beamforming*, não implica que a técnica não possa ser considerada para aplicações em redes sem fio mais generalistas. Ela poderia ser utilizada, por exemplo, como base para sistemas de comunicação cooperativa entre grupos de elementos de uma rede genérica. Nesse contexto, seria interessante investigar que impacto o uso de *beamforming* distribuído causaria nas camadas superiores dos protocolos de comunicação.

A busca por métodos com menor latência, idealmente com apenas dois *time-slots*, oferece oportunidade para desenvolvimentos que poderiam resultar em sistemas com maior escalabilidade do que a dos sistemas investigados até então, conforme discutido no Apêndice B.

E, por fim, mas não menos importante, análise e quantificação dos efeitos de erros de fase nos ganhos de *beamforming*, bem como considerações quanto a robustez do sistema quando considerados os efeitos do número de sensores e a existência de interferências.

Apêndice A

Implementação

Neste capítulo, após a apresentação da infraestrutura de SDR utilizada nessa tese, é descrito o arcabouço construído sobre a plataforma de *software* livre *GNU-Radio* e utilizado na implementação dos receptores e na validação experimental do método *Inversão Remota de Fase* (IRF), bem como na implementação de demonstração dos métodos *Time-Slotted Round-Trip* (RT) e *One Bit Feedback* (OBF), descritas a seguir.

A.1 Plataforma Experimental SDR

A plataforma de rádio definido em software utilizada no arranjo experimental para essa tese é composta de dois elementos que se complementam: Uma ferramenta de software (GNU-Radio) [62] e uma interface de hardware, Universal Software Radio Peripheral (USRP) [63].

A.1.1 *Software*

GNU-Radio é uma ferramenta de software gratuita e de código aberto, composta de um conjunto de funções de processamento de sinais, utilizada na implementação de Rádios Definidos por Software. A exemplo de outras plataformas de software, GNU-Radio utiliza uma arquitetura baseada em camadas para o desenvolvimento de aplicações. O núcleo principal é implementado na forma de blocos primitivos de processamento de sinais e está escrito em C++ visando velocidade e robustez, enquanto que as interfaces e os controles

de configuração e fluxo são implementadas em Python. Ela também pode ser vista como uma coleção de funções de processamento de sinais escritas em C++ que são gerenciadas por uma camada de software escrita em Python. O código Python, essencialmente cria grafos, *flowgraphs*, que controlam o fluxo de informações entre blocos de código que, por sua vez, implementam funções de processamento de sinais. Esses grafos conectam blocos produtores de informação (sources) a blocos consumidores (sinks) representando a dinâmica da aplicação. Os scripts Python criados pelo usuário são responsáveis por criar, executar e gerenciar os grafos. Esta arquitetura está ilustrada, de forma simplificada, na Figura A.1.

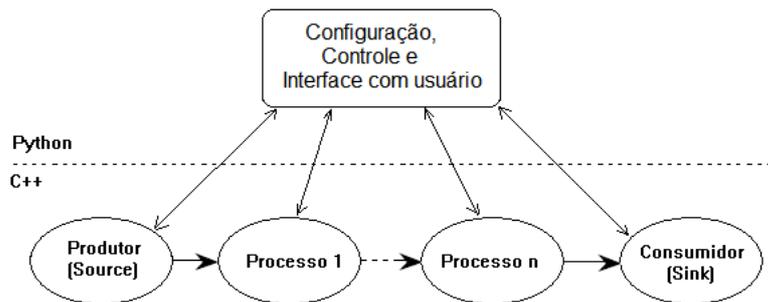


Figura A.1: Estrutura de aplicações desenvolvidas em GNU-Radio.

A.1.2 Hardware

A plataforma USRP é uma das plataformas de hardware mais usuais para projetos de SDR utilizando GNU-Radio. Essa plataforma permite que computadores de uso geral se comportem como rádios definidos em software operando em alta velocidade. Ela contém conversores analógico/digitais de alta velocidade (ADC a 100 MS/s e DAC a 400 MS/s, no caso do modelo utilizado nessa tese) e um dispositivo FPGA de alta capacidade para implementação de filtros digitais e conversores de frequência (*upconverter* e *downconverter*), bem como interface *gigabit ethernet* para comunicação com o computador hospedeiro da aplicação, que a abstrai como uma interface de rede capaz de processar amostras complexas (I/Q) em bandabase tanto no sentido de transmissão (computador para USRP) como no sentido de recepção (USRp para computador). A comunicação de RF é implementada na forma de placas de interface de RF disponíveis em diferentes bandas de frequência. Na parte experimental dessa tese, foi utilizado o modelo USRP-N2x0 com placas de RF modelo

RFX900, operando na faixa de frequências de 902 MHz a 928 MHz. Um diagrama de blocos representando a combinação do módulo USRP N210 com a interface analógica RFX900 está mostrado na Figura A.2.

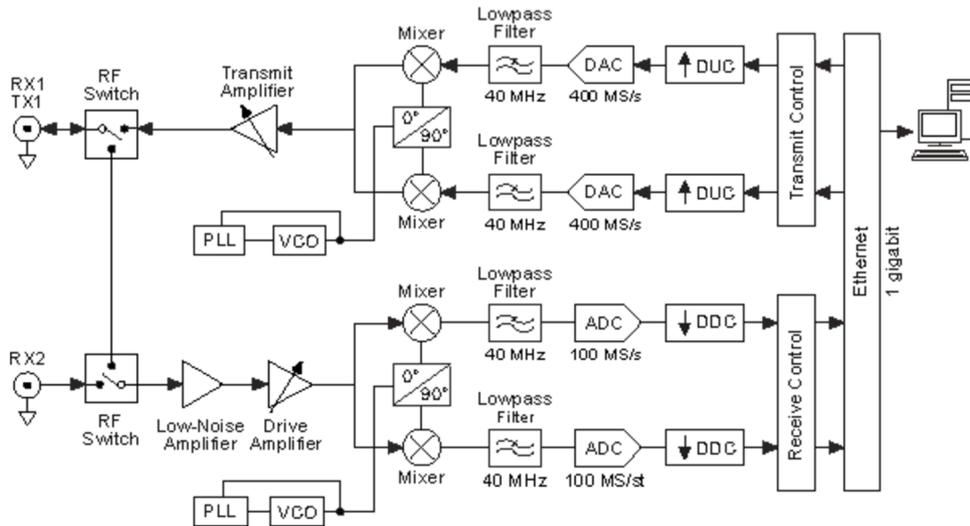


Figura A.2: Diagrama de blocos da plataforma de hardware USRP N2x0. Fonte: reproduzido de [63].

A.2 Arcabouço

O arcabouço criado para essa tese é uma coleção de blocos OOT (*Out-of-Tree*), que na terminologia do GNU-Radio, representam funções de processamento de sinais definidas e criadas pelo usuário. Esses blocos executam funções específicas que não existem na biblioteca original da ferramenta, como defasadores, por exemplo, ou funções que facilitam a implementação de aplicações que operam em modo Half-Duplex ou Time Division Duplex (TDD) como o bloco *mute* que silencia o transmissor para que não haja auto-interferência durante o processo de recepção. Alguns desses blocos estão mostrados Figura A.3.

Para essa tese em particular, o principal elemento é o bloco *My fsm* que implementa um controlador sequencial baseado em Máquina de Estados Finitos do tipo Moore. Sua função principal é controlar a sequência de operações, ativando e desativando os outros blocos de acordo com o algoritmo pretendido. A sequência de operações é especificada por uma tabela de transição de estados codificada em um arquivo texto que é lido e interpretado pelo bloco no início da execução do sistema.

Segue uma descrição funcional de alguns blocos:

- **Sig Det:** *Detector simples de sinal baseado no cálculo da média móvel do módulo do sinal de entrada, guiado por um limiar parametrizado ('threshold').*
- **My hold:** *Controle 'Sample and Hold', que memoriza e mantém em sua saída o valor presente na entrada quando da transição negativa do sinal 'hold'.*
- **My muxen:** *Seletor, direcionando para a saída o sinal da entrada 1 quando o sinal 'sel' for igual a 1 ou o sinal da entrada 2 quando 'sel' for igual a 2 ou 'zero' (mute) quando 'sel' for igual a zero.*
- **Phase Shifter:** *Implementa, de forma contínua, um deslocador de fase calculando $out = in * e^{j(g*phase)}$*
- **My dpll_hold:** *PLL digital que rastreia a frequência e a fase do sinal de entrada e o apresenta na saída quando o sinal 'hold' for igual a 1 e interrompe o rastreamento da entrada mantendo, na saída, um sinal com frequência e fase iguais ao do sinal de entrada quando da transição negativa do sinal 'hold'. Como conveniência, também fornece a diferença de fase entre o sinal de entrada e um sinal local de referência.*
- **My fsm:** *Controlador sequencial programável, baseado em máquina de estados finitos (FSM) tipo Moore, cuja tabela de transições de estados é interpretada em tempo real, a partir de um arquivo texto.*

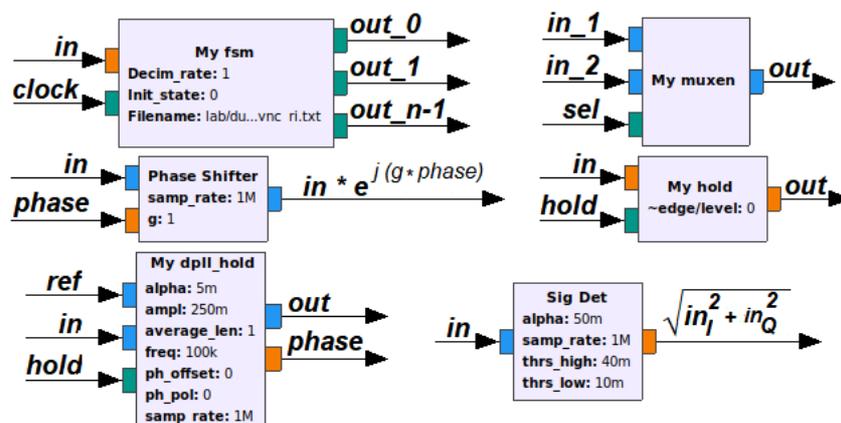


Figura A.3: Blocos básicos do arcabouço.

A.3 Implementação dos Blocos Principais

Para facilitar a construção de blocos OOT, o GNU-Radio disponibiliza uma ferramenta com todos os arquivos necessários para a compilação de funções, escritas em C++, compatíveis com sua estrutura interna. Essa ferramenta disponibiliza, entre outros arquivos auxiliares, um modelo de código pronto com apenas uma função a ser completada pelo usuário: a função *work*. A comunicação da função com o GNU-Radio se dá através de *buffers* de entrada e de saída. As amostras de entrada, oriundas de outros blocos, são disponibilizadas nos *buffers* de entrada. A função *work* deve acessar essas amostras, processá-las como for conveniente e as amostras resultantes desse processamento devem ser armazenadas no(s) *buffer(s)* de saída, para serem entregues, pelo escalonador, para o(s) bloco(s) subsequente(s). Todo o processo é sincronizado pelo escalonador com base na taxa de amostragem do sistema.

Todos os blocos obedecem o mesmo padrão de codificação, com diferenças apenas na função específica que implementam.

Como ilustração, um trecho de código do bloco *Phase shifter*, obedecendo esse padrão, está mostrado na Listagem A.3.1.

A.3.1 Deslocador de Fase (*Phase_shifter*)

Esse bloco recebe dois fluxos de dados (amostras) em dois buffers de entrada. O primeiro buffer contém as amostras do sinal complexo a ter sua fase modificada pelo fator presente no segundo buffer.

O código, mostrado na Listagem A.3.1, executa um *loop* varrendo cada posição dos buffers de entrada, convertendo o valor da *phase*, presente no segundo buffer, para um número complexo, e em seguida multiplicando esse número pela amostra presente no primeiro buffer. O resultado é armazenado no buffer de saída e representa a expressão $out = in * e^{j(phase)}$.

```

1      // WORK_FUNCTION (definida pelo usuario).
2      int my_shifter_impl::work(int noutput_items,
3      gr_vector_const_void_star &input_items,
```

```

4     gr_vector_void_star &output_items)
5     {
6         const gr_complex* in0 = (gr_complex*)input_items[0];
7         const float* phase = (const float *) input_items[1];
8         gr_complex* out = (gr_complex*)output_items[0];
9         int size = noutput_items;
10        float t_real, t_imag;
11        while (size-- > 0) {
12            gr::sincosf(*phase++, &t_imag, &t_real);
13            *out++ = *in0++ * gr_complex(t_real, t_imag);
14        }
15        return noutput_items;
16    }

```

Listagem 4.1: Trecho de código do bloco *Phase shifter*

A.3.2 PLL Digital (*DPLL_hold*)

Esse bloco, construído na forma de um bloco hierárquico composto de outros blocos, tem três portas de entrada e duas portas de saída. Como entradas ele recebe o sinal recebido pelo rádio *rx*, o sinal de um oscilador de referência *ref_osc* e um sinal de controle *hold_pll*, como se pode ver na Figura A.4.

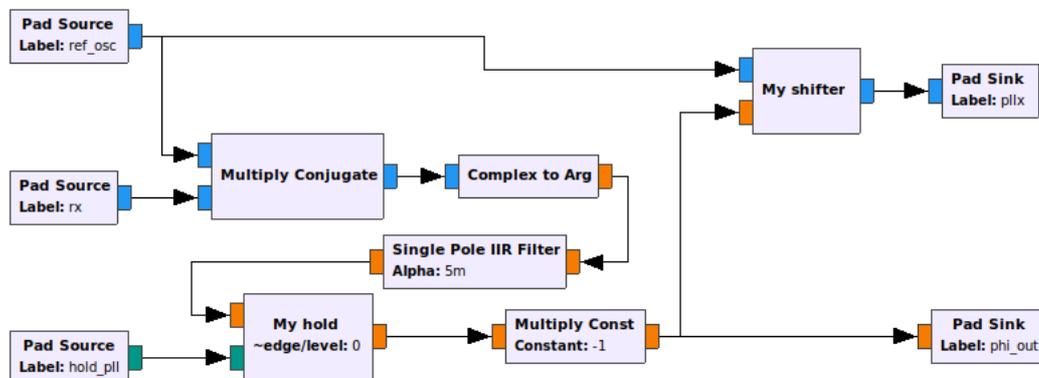


Figura A.4: Diagrama de blocos hierárquico da implementação do PLL.

Considerando que existe sincronismo de frequência, esse subsistema mede a diferença de fase existente entre o sinal de referência e o sinal recebido. Essa diferença de fase é suavizada

por um filtro IIR de primeira ordem e, em seguida, armazenada pelo bloco *my_hold* no momento que o sinal de controle *hold_pll* é ativado. A saída do bloco *my_hold* é usada, a partir desse momento, como fator para correção de fase do sinal de referência pelo bloco *my_shifter*. O resultado desse processo é que o sinal de saída, *plx*, é um sinal com fase igual à fase do sinal recebido e que se mantém estável mesmo na ausência do sinal *rx*. Por conveniência, a diferença de fase é disponibilizada na saída *phi_out*.

A.3.3 Controlador (FSM)

Durante a inicialização do sistema, esse bloco lê um arquivo texto contendo uma representação textual de uma tabela de transição de estados a ser percorrida durante a execução do sistema. As informações são transformadas de texto para um formato mais apropriado, sendo armazenadas em um vetor bi-dimensional em que as linhas representam os estados e as colunas representam as saídas e os números dos possíveis próximos estados.

A tabela de estados é representada por um arquivo texto com o seguinte formato:

```
# Essa linha é apenas comentário.....
ss:o0, o1, o2, ..., on-1:ps0/ps1; comentário...
```

com a seguinte interpretação:

```
ss - estado corrente ou atual (00, 01, 02, ...N-1).
o0, o1,... on-1 - saídas (saída 0, saída 1, ...saída n-1).
ps0 - próximo estado caso a entrada seja igual a zero.
ps1 - próximo estado caso a entrada seja maior que zero.
; - tudo que estiver após o caractere ';' é considerado comentário.
[ ] - espaços são ignorados.
```

O bloco recebe dois sinais, *clock* e *in*. O sinal *clock* é sincronizado com a taxa de amostragem do sistema e determina a velocidade de mudança dos estados. A cada mudança do *clock*, os valores registrados na tabela de transição de estados correspondentes ao estado corrente são apresentados nos terminais de saída. Em seguida, o sinal *in* é testado e seu valor é usado para determinar o próximo estado a ser executado na próxima mudança de *clock*.

O número de estados, N , é um parâmetro configurável e se o valor do campo $ps0$ for maior que N , um temporizador interno é ativado, forçando uma pausa de $(ps0 - N)$ mudanças de *clock* antes de seguir para o próximo estado, que obrigatoriamente será o estado $ps1$.

Com objetivo de ilustrar o uso da máquina de estados no controle de uma aplicação *Half-Duplex*, considere um receptor que, após sincronizar seu PLL com um sinal piloto recebido de um interrogador, responde com um sinal de mesma frequência e fase. O diagrama de blocos, representando o *grafo* dessa aplicação (*transponder*), está apresentado na Figura A.5.

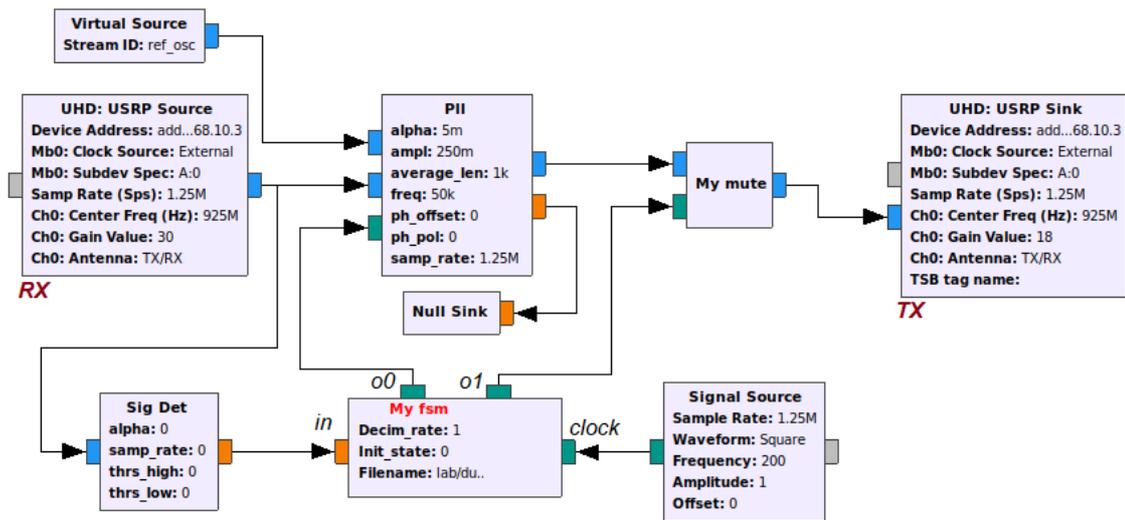


Figura A.5: Diagrama de blocos de aplicação ilustrativa (*transponder*).

O sequenciamento dessa aplicação pode ser implementado por uma tabela de estados representada pela seguinte listagem (considerando que o sistema tenha sido configurado para um limite máximo de $N = 50$ estados):

```
#
# sequência para aplicação exemplo (transponder)
#
# ss: pll, tx :ps0/ps1
#
00: 0, 0 : 00/01; se sinal detectado, estado 01, senão, aguardar
```

01: 1, 0 : 54/02; habilitar o PLL por 4 ciclos de clock, continuar
 02: 0, 0 : 03/03; travar o PLL e continuar
 03: 0, 0 : 04/03; se sinal ainda presente, aguardar, senão, continuar
 04: 0, 1 : 58/05; ativar transmissor por 8 ciclos de clock, continuar
 05: 0, 0 : 00/00; desativar transmissor e retorna para o estado 00

A.4 Implementação dos Interrogadores

- *Transceptor Interrogador (D) para o método One Bit Feedback (OBF)*

A implementação do interrogador para o método *One Bit Feedback* (OBF) pode ser representada pelo diagrama de blocos mostrado na Figura A.6 e sua funcionalidade pode ser representada pelo pseudo-código Algoritmo 1.

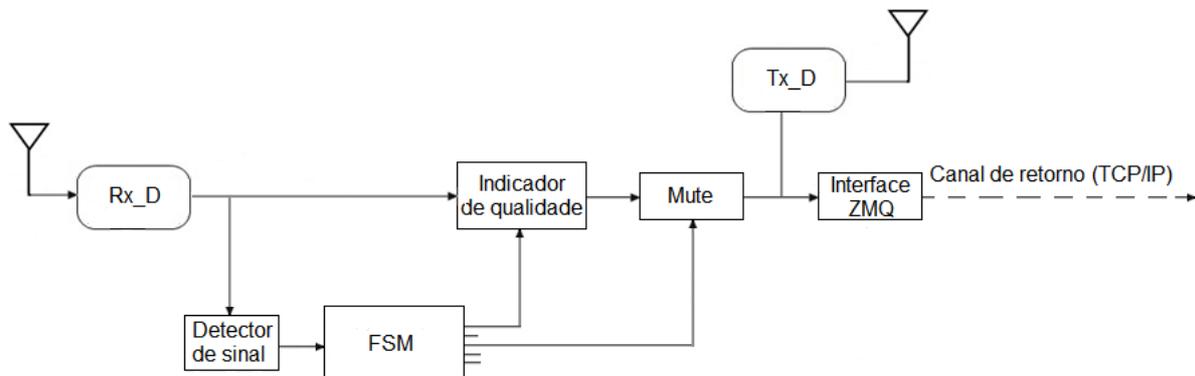


Figura A.6: Diagrama de blocos do interrogador para o método OBF.

Como o aspecto investigado nessa tese é apenas o método de alinhamento de fase, a técnica ou forma de modulação utilizada para transmitir o sinal de retorno é considerada irrelevante, e para simplificar, a transmissão de retorno do interrogador para os sensores é implementada via protocolo TCP/IP pelo bloco ZQM. O interrogador é composto de um receptor, um detector de sinal, um controlador sequencial e um bloco de comunicação ZMQ. O bloco ZQM é um bloco digital para comunicação que utiliza o protocolo TCP/IP.

O detector de sinal é continuamente monitorado pelo controlador (fsm) e a indicação

de presença sinal força o controlador a ativar o avaliador de qualidade de sinal. Esse nível de sinal é comparado com o nível avaliada no *time slot* anterior. O resultado dessa comparação é armazenado temporariamente e o controlador passa aguarda o fim do sinal recebido. Quando essa condição é estabelecida, o controlador ativa o bloco ZQM para que a informação de qualidade quantizada seja enviada para os sensores e ativa o transmissor por um curto intervalo de tempo para que os sensores detectem um sinal indicando que o sinal de retorno válido está presente no bloco ZQM. Em seguida o controlador volta para o estado inicial de monitoração do sinal recebido. O uso do protocolo TCP/IP garante a correta recepção do sinal de realimentação pelos sensores emulando o canal garantido de retorno como requerido pelo protocolo original.

Esse processo está representado na forma de pseudo-código no Algoritmo 1.

Algoritmo 1: Transceptor do Interrogador (D) para o método OBF

início

inicialização;

$BIT = 0$;

$RSSI_anterior = 0$;

repita

repita

avaliar *detector_de_sinal*;

até *sinal_presente*;

ligar medidor de *RSSI*;

if $RSSI > RSSI_anterior$ **then**

$BIT = 1$;

else

$BIT = 0$;

end

repita

avaliar *detector_de_sinal*;

até *sinal_ausente*;

desligar medidor de *RSSI*;

enviar *BIT* via *TCP/IP*;

até *fim da campanha*;

fim

- *Transceptor Interrogador (D) para os métodos Inversão Remota de Fase (IRF) e Time-Slotted Round-Trip (RT)*

A implementação do interrogador para os métodos *Inversão Remota de Fase (IRF)* e *Time-Slotted Round-Trip (RT)* pode ser representada pelo diagrama de blocos mostrado na Figura A.7 e sua funcionalidade representada pelo pseudo-código mostrado no Algoritmo 2. Sua operação se resume a transmitir um sinal piloto e a recepção do sinal transmitido pelos sensores. Inicialmente, o controlador habilita o oscilador de referência e ativa o transmissor por um determinado intervalo de tempo. Em seguida o transmissor é desativado e o controlador passa a monitorar o detector de sinal. A presença de sinal informada pela detector ativa a sequência, no controlador, para indicar o nível do sinal recebido. em seguida aguarda o final desse sinal para reiniciar esse processo.

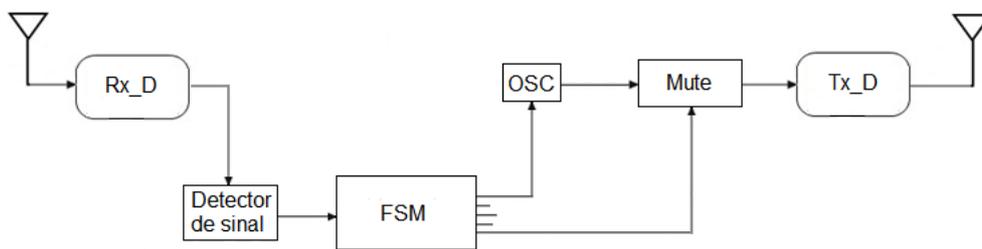


Figura A.7: Diagrama de blocos do interrogador para os métodos RT e IRF.

Algoritmo 2: Transceptor do Interrogador (D) para os métodos IRF e RT

início

inicialização;

repita

ligar transmissor;

transmitir sinal_piloto;

desligar transmissor;

repita

avaliar *detector_de_sinal*;

até *sinal_presente*;

ligar medidor de *RSSI*;

repita

avaliar *detector_de_sinal*;

até *sinal_ausente*;

desligar medidor de *RSSI*;

até *fim da campanha*;

fim

A.5 Implementação dos Transceptores

- *Transceptores para o método One Bit Feedback (OBF)*

A implementação dos transceptores para o método *One Bit Feedback* (OBF) pode ser representada pelo diagrama de blocos mostrado na Figura A.8 e sua funcionalidade representada pelo pseudo-código mostrado no Algoritmo 3.

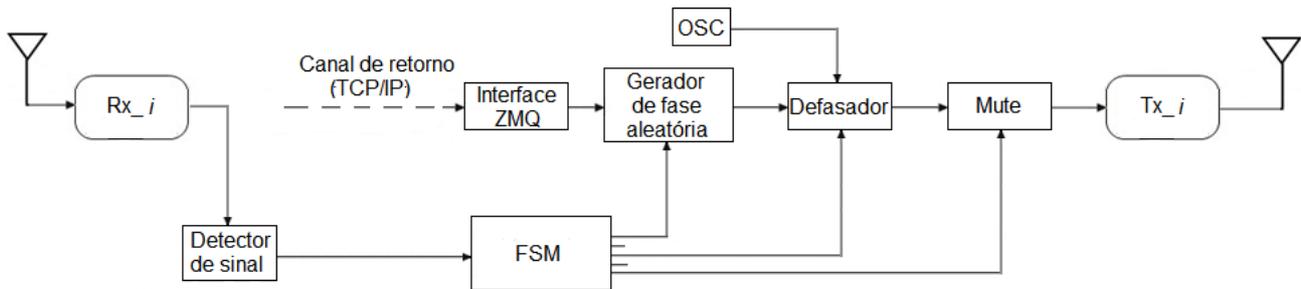


Figura A.8: Diagrama de blocos dos transceptores para o método OBF.

Algoritmo 3: Transceptor dos Sensores para o método OBF

início

inicialização;

$FASE = 0$;

$FASE_anterior = 0$;

repeat

repeat

avaliar *detector_de_sinal*;

until *senal_presente*;

if $BIT_recebido == 1$ **then**

$FASE_anterior = FASE$;

$FASE = FASE + RND()$;

else

$FASE = FASE_anterior$;

$FASE = FASE + RND()$;

end

habilitar correção de fase ;

habilitar transmissor e transmitir *SINAL_tom*;

until *fim da campanha*;

fim

- *Transceptores para o método Time-Slotted Round-Trip (RT)*

A implementação dos transceptores para o método *Time-Slotted Round-Trip* (RT), pode ser representada pelo diagrama de blocos mostrado na Figura A.9 e sua funcionalidade representada pelo pseudo-código mostrado no Algoritmo 4.

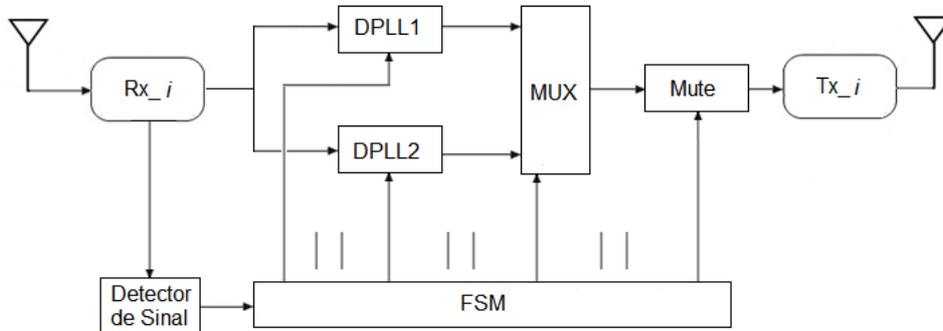


Figura A.9: Diagrama de blocos dos transceptores para o método RT.

Inicialmente, o controlador monitora o detector de sinal e, ao detectar a presença de sinal, o que caracteriza o início do *time-slot* 0, habilita o PLL₁ por um tempo suficiente para que o PLL rastreie a fase do sinal recebido. Em seguida o controlador aguarda até que o sinal recebido desapareça, quando então comanda o multiplexador para habilitar o sinal do PLL₁ em sua saída e ativa o transmissor por um determinado intervalo de tempo (*time-slot* 1), efetivamente retransmitindo o sinal recebido do interrogador. O transmissor é, então desativado e o controlador volta a monitorar o detector de sinal aguardando pelo sinal de retransmissão do outro sensor. Quando essa condição é detectada, *time-slot* 2, o controlador habilita o PLL₂ que rastreia a fase do sinal retransmitido pelo outro sensor. Novamente o controlador monitora o detector de sinal até que a retransmissão do outro sensor seja finalizada. Nesse momento o controlador comanda o multiplexador para selecionar o sinal do PLL₂ e ativa o transmissor por um determinado intervalo de tempo, *time-slot* 3, transmitindo para interrogador o sinal recebido do outro sensor.

Esse processo está representado na forma de pseudo-código no Algoritmo 4.

Algoritmo 4: Transceptor dos Sensores para o método RT

```

início
  inicialização;
  repeat
    repeat
      avaliar detector_de_sinal;
    until sinal_presente;
    habilitar PLL_1;
    repeat
      avaliar detector_de_sinal;
    until sinal_ausente;
    selecionar (mux) e habilitar transmissor;
    transmitir PLL_1;
    repeat
      avaliar detector_de_sinal;
    until sinal_presente;
    habilitar PLL_2;
    repeat
      avaliar detector_de_sinal;
    until sinal_ausente;
    selecionar (mux) e habilitar transmissor;
    transmitir PLL_2;
  until fm da campanha;
fim

```

- *Transceptores para o método Inversão Remota de Fase (IRF) com fase conhecida*

A implementação dos transceptores para o método proposto (IRF) com fase conhecida, pode ser representada pelo diagrama de blocos mostrado na Figura A.10.

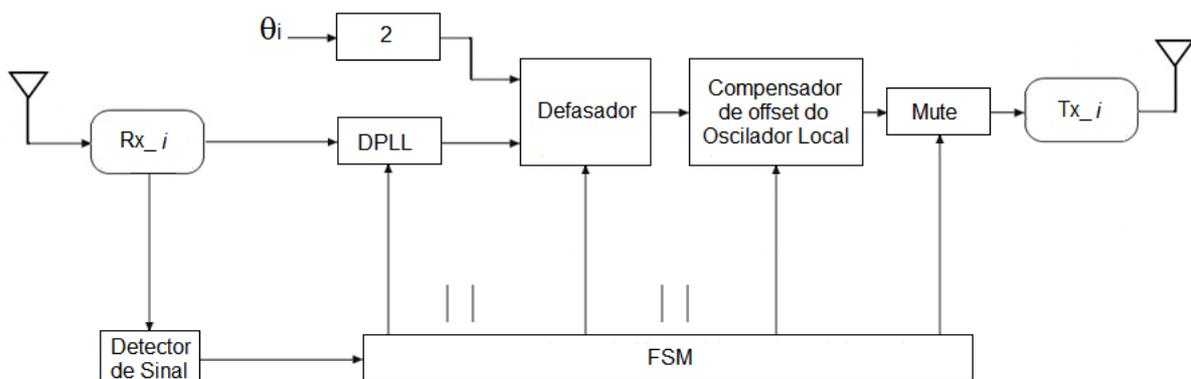


Figura A.10: Diagrama de blocos dos transceptores para o experimento IRF com fase conhecida.

Ao detectar o sinal de *beacon* enviado pelo interrogador, o controlador ativa o PLL que se sincroniza com o sinal de *beacon*, e, assumindo que ele tem a forma $\cos(\omega t + \phi_{tx_D} - \theta_i - \phi_{rx_i})$, ativa o bloco *defasador* para gerar o sinal $\cos(\omega t + \phi_{tx_D} + \theta_i - \phi_{rx_i})$, efetivamente invertendo θ_i . Em seguida, para compensar o efeito das fases iniciais dos osciladores locais, o *compensador de offset* do OL acrescenta uma rotação adicional de fase de $\phi_{rx_i} - \phi_{tx_i}$, determinada previamente durante o estágio de calibração. O transmissor é, então, ativado, transmitindo o sinal $\cos(\omega t + \phi_{tx_D} + \theta_i)$ de volta para o interrogador.

Sua dinâmica é representada pelo pseudo-código mostrado no Algoritmo 5.

Algoritmo 5: Transceptor dos Sensores para o método IRF com fase conhecida

início

```

inicialização (TETA, TAU);
##### CALIBRAÇÃO #####
selecionar NCO (mux);
habilitar transmissor por TAU segundos;
repeat
    avaliar detector_de_sinal;
until sinal_presente;
habilitar PLL (modo de rastreamento);
travar PLL;
armazenar PSI (offset do oscilador local);
repeat
    avaliar detector_de_sinal;
until sinal_ausente;
##### OPERAÇÃO #####
repeat
    repeat
        avaliar detector_de_sinal;
until sinal_presente;
habilitar PLL (modo de rastreamento);
travar PLL;
repeat
        avaliar detector_de_sinal;
until sinal_ausente;
aplicar subtração de TETA;
compensar offset do oscilador local com PSI;
selecionar PLL (mux);
habilitar transmissor por TAU segundos;
until fim da campanha;

```

fim

- *Transceptores para o Método Inversão Remota de Fase (IRF)*

A implementação dos transceptores para o método proposto (IRF), pode ser representada pelo diagrama de blocos mostrado na Figura A.11 e sua efetiva implementação em GNU-Radio está mostrada na Figura A.12, cuja dinâmica é capturada pelo fluxograma mostrado na Figura A.13. Durante o estágio de calibração, a fase do sinal de retorno do mestre é capturada pelo bloco tracejado na Figura A.12. Posteriormente, durante o estágio de operação, o sinal piloto enviado pelo interrogador tem sua fase invertida pelo bloco denominado *Phase_inverter* e sofre uma rotação adicional de fase pelo bloco *Phase_rotator*.

Considerando a fase de operação, ao detectar a presença de sinal, o controlador habilita o PLL por um tempo suficiente para que sua saída reproduza o sinal piloto recebido. Em seguida o controlador, sequencialmente, ativa o inversor de fase, ativa o compensador de fase e ativa o transmissor por um determinado intervalo de tempo. Após esse intervalo o controlador retorna ao estado inicial, quando volta a monitorar o detector de sinal à espera de um novo sinal piloto enviado pelo interrogador.

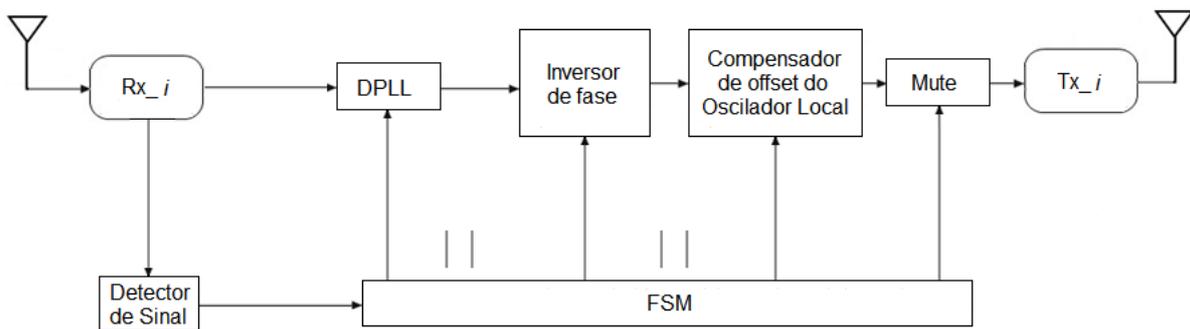


Figura A.11: Diagrama de blocos dos transceptores para o método IRF.

Se esse for o transceptor do sensor S_0 (mestre), sua operação é mais simples: Ao detectar a presença de sinal, o controlador simplesmente habilita o PLL para reproduzir o sinal piloto e ativa o inversor de fase, mantendo o compensador de fase desabilitado. O transmissor é ativado por um determinado intervalo de tempo e retorna ao estado inicial.

A sequência de passos executados pelo controlador é representada pseudo-código mostrado no Algoritmo 6.

Algoritmo 6: Transceptor dos Sensores para o método IRF

```

início
  inicialização;
  #;
  # CALIBRAÇÃO;
  #;
  selecionar NCO (mux);
  habilitar transmissor;
  desabilitar transmissor;
  repeat
    avaliar detector_de_sinal;
  until sinal_presente;
  habilitar PLL (modo de rastreamento);
  travar PLL;
  armazenar PSI (offset do oscilador local);
  repeat
    avaliar detector_de_sinal;
  until sinal_ausente;
  #;
  # OPERAÇÃO;
  #;
  repeat
    repeat
      avaliar detector_de_sinal;
    until sinal_presente;
    habilitar PLL (modo de rastreamento);
    travar PLL;
    repeat
      avaliar detector_de_sinal;
    until sinal_ausente;
    inverter fase do sinal do PLL;
    compensar offset do oscilador local com PSI;
    selecionar PLL (mux);
    habilitar transmissor;
    desabilitar transmissor;
  until fim da campanha;
fim
  
```

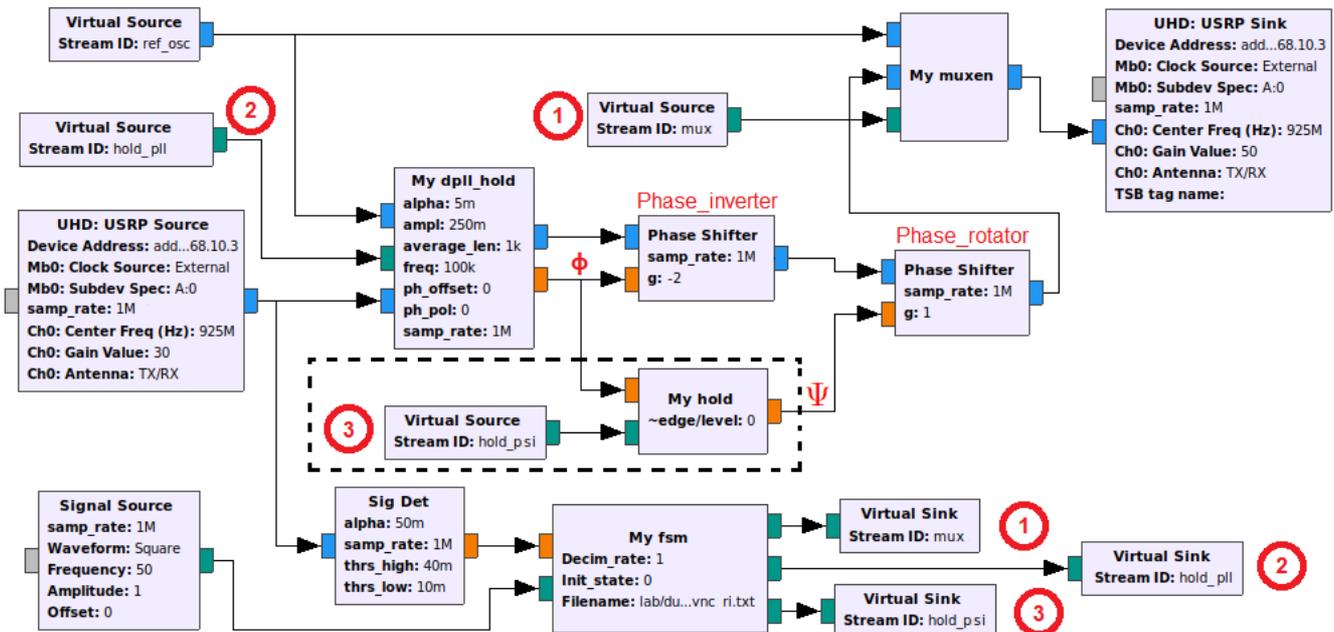


Figura A.12: Diagrama esquemático (*grafo*) da implementação dos transceptores em GNU-Radio para o método proposto. A parte tracejada é mantida inativa no modo Mestre.

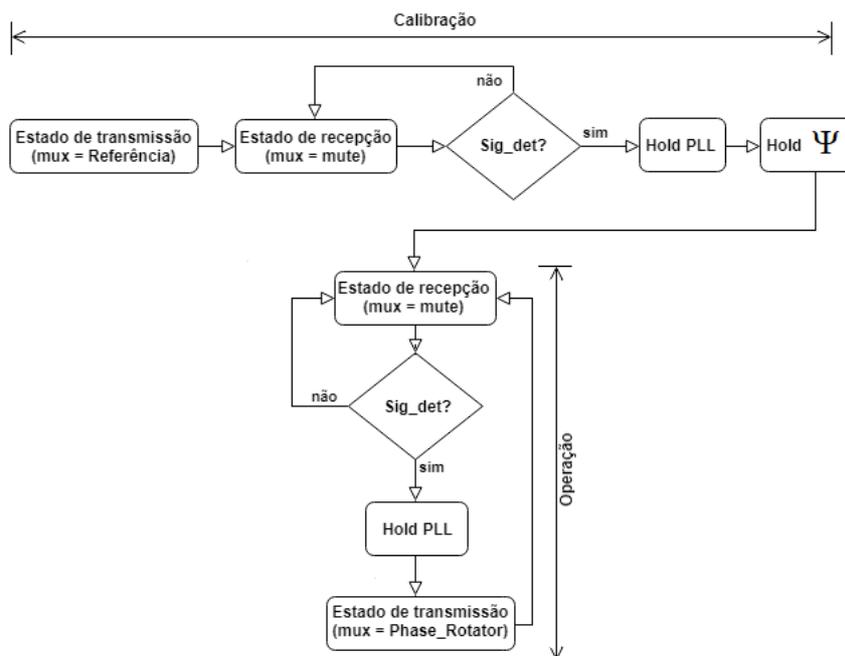


Figura A.13: Fluxograma representando a dinâmica do método IRF. O Mestre ignora o estágio de calibração, iniciando diretamente no estágio de operação.

A.6 Conclusão

Neste capítulo, foram apresentadas e descritas as implementações dos transceptores sobre a plataforma GNU-Radio, bem como foram apresentados os elementos componentes do arcabouço criado especificamente para essa tese.

Apêndice B

Método Alternativo

No método *Timed-Slotted Round Trip* o tempo de resposta (latência) do conjunto de sensores para o interrogador depende do número de sensores e é distribuído por toda a campanha de testes. Já no método *Inversão Remota de Fase*, a latência é concentrada na fase de calibração e é minimizada na fase de operação do sistema mas, ainda assim, é inconveniente e motivou uma investigação na busca de um método que não dependesse do número de sensores. Essa investigação resultou no método *Inversão de Fase baseada em Duplo Tom* (IFDT).

B.1 Inversão de Fase baseada em Duplo Tom (IFDT)

Como discutido no Capítulo 3, a fase de chegada do sinal enviado pelo interrogador e recebido pelo elemento sensor é composta de duas partes: uma parte relacionada ao percurso e uma parte provocada pelo processo de conversão de frequência (*downconverter*), que depende diretamente da fase inicial do oscilador local do receptor.

Assumindo que o deslocamento de fase adicionado pelo *downconverter* no processo de recepção, pode ser cancelado pelo *upconverter* no processo de transmissão, apenas a parte da fase correspondente ao percurso deve ser invertida, e para isso é necessário separar a componente de fase correspondente ao oscilador local. Como a fase inicial do oscilador local não é conhecida, tem-se um problema com duas variáveis desconhecidas e apenas uma medida da combinação dessas variáveis.

O método proposto para estimação e inversão dessa fase de chegada se baseia na transmissão, pelo interrogador, de um um sinal composto de duas frequências distintas, que resultam em duas medidas diferentes de fase de chegada do sinal. Essas duas medidas contêm, cada uma, uma parte variável (percurso) que é diferente para cada medida, e uma parte constante (oscilador local) comum às duas medidas. Dessa forma é possível estimar a parte da fase relativa ao percurso e invertê-la, emulando o mecanismo utilizado pelos arranjos retrodiretivos.

O mecanismo proposto para a estimação de fase é similar à técnica *Multi-carrier Phase Difference of arrival* (PDoA), utilizada em radares e sistemas de localização [69]-[73]. Entretanto em vez ser utilizada para estimar posição, nessa tese, a técnica é utilizada para estimar a fase de chegada do sinal.

Tomando-se como referência a Seção 3.2, a real complexidade do método de inversão de fase está, justamente na estimação do ângulo θ_i . O ângulo de fase visto pelo processamento em banda base é, na verdade, $\gamma_i = -\theta_i - \phi_i$ e ϕ_i é, em geral, desconhecido.

A solução proposta é baseada na observação de que, para uma distância fixa, o deslocamento de fase sofrido por um sinal está linearmente relacionado com sua frequência [74]. Conforme ilustrado na Figura B.1, após percorrer uma certa distância, um sinal de referência com frequência f_1 sofre um deslocamento de fase de $\theta_1 = \pi/4 \text{ rad}$, enquanto que um sinal com frequência $f_2 = 2f_1$, percorrendo a mesma distância, sofre um deslocamento de fase de $\theta_2 = 2\theta_1 = \pi/2 \text{ rad}$.

Se o sinal de *beacon* for composto de dois segmentos com frequências f_1 e f_2 , respectivamente, pode-se derivar um sistema simples de duas equações como

$$-\theta_i^1 - \phi_i = \gamma_i^1 \quad (\text{B.1})$$

$$-\theta_i^2 - \phi_i = \gamma_i^2 \quad (\text{B.2})$$

sendo θ_i^1 o deslocamento de fase sofrido pelo segmento de frequência f_1 no sinal recebido

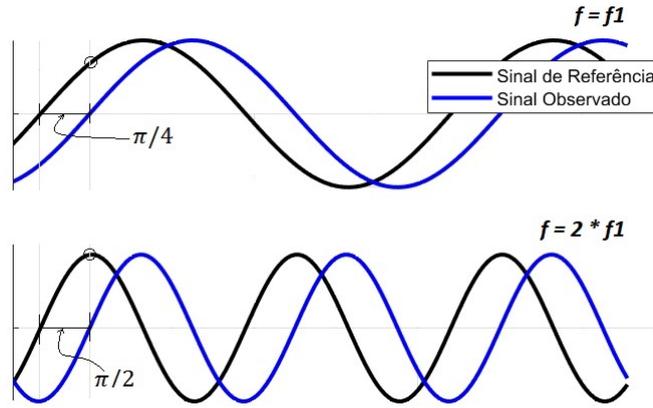


Figura B.1: Relação Linear entre deslocamento de fase e frequência.

em S_i devido ao percurso e θ_i^2 o deslocamento de fase sofrido pelo segmento de frequência f_2 no mesmo percurso.

Fazendo $f_2 = kf_1$, com $k > 1$, garante-se $\theta_i^2 = k\theta_i^1$, e subtraindo a Equação (B.2) da Equação (B.1), tem-se

$$-\theta_i^1 + k\theta_i^1 = \gamma_i^1 - \gamma_i^2, \quad (\text{B.3})$$

que permite obter uma estimativa de θ_i^1 , independentemente da fase inicial (desconhecida) do oscilador local, ϕ_i , e o resultado é a relação

$$\hat{\theta}_i^1 = \frac{\gamma_i^1 - \gamma_i^2}{k - 1}, \quad (\text{B.4})$$

que representa a estimativa de fase procurada, considerando o segmento de sinal com frequência f_1 , e que pode ser invertida, isoladamente, independente de ϕ_i , gerando $tbb_i(t) = \cos(2\pi f_1 t + \hat{\theta}_i - \phi_i)$, Equação (3.4), a partir de $rbb_i(t) = \cos(2\pi f_1 t - \theta_i - \phi_i)$, Equação (3.3).

Os ângulos γ_i^1 e γ_i^2 são resultados de filtros de média móvel aplicados aos ângulo instantâneos desdobrados de fase (*phase unwrapping*) que, por sua vez, são obtidos da relação $\gamma_{IQ} = \arctan(Q/I)$, calculada sobre os sinais I (*Inphase*) e Q (*Quadrature*) disponíveis na saída do *downconverter*.

B.2 Possível implementação do método IFDT

O sinal de duplo tom recebido pelos sensores é filtrado para recuperar os segmentos f_1 e f_2 . Como não faz muito sentido a comparação de fase entre sinais de frequências diferentes, ambos os tons são convertidos digitalmente para uma frequência intermediária comum $f_{1_{IF}} = f_{2_{IF}} = 50$ KHz. Suas fases desdobradas são representadas por $\gamma_i = -(\theta_i + \phi_i)$ nas Equações (B.1) e (B.2), de onde a diferença desdobrada de fase é calculada e multiplicada pelo fator $1/(k-1)$ gerando $\hat{\theta}_1$. Esse valor é duplicado e aplicado ao deslocador final de fase, efetivamente invertendo a fase do canal.

A determinação da relação entre as frequências f_1 e f_2 , deve levar em consideração as limitações de largura de banda do hardware, os níveis de quantização dos conversores A/D e o número de bits do downconverter da plataforma de hardware específica.

Esse método foi implementado de acordo com o diagrama de blocos mostrado na Figura B.2, com uma particularidade: O algoritmo assume que existe um oscilador local comum compartilhado pelos circuitos de transmissão e recepção, de forma que as fases iniciais do transmissor e do receptor se cancelam durante os processos de *downconversion* e *upconversion*.

Entretanto, as interfaces de RF utilizadas no arranjo experimental para essa tese utilizam dois PLLs independentes, um para cada sentido da comunicação e não há como evitar a diferença de fase constante mas aleatória que se forma entre eles. Para contornar o problema, foi necessária a implementação de um método de calibração automática para, virtualmente, equalizar as fases dos osciladores locais.

A interface de RF, *RFX900*, dispõe de duas conexões diferentes para antenas, uma para transmissão/recepção em aplicações *Time – Division Duplex* (TDD) chamada *RX/TX* e outra apenas para recepção chamada de *RX2*.

Nesta aplicação, em particular, a conexão da antena *RX/TX* é usada como porta principal de RF e a conexão *RX2* é usada como porta de realimentação local.

O processo de calibração (não mostrado na Figura B.2) é executado em dois passos no começo de cada campanha de testes:

No *passo1* um sinal de calibração $cal_{tx}(t) = \cos(\omega t)$, após a conversão para RF e transmitido pela antena *RX/TX* como

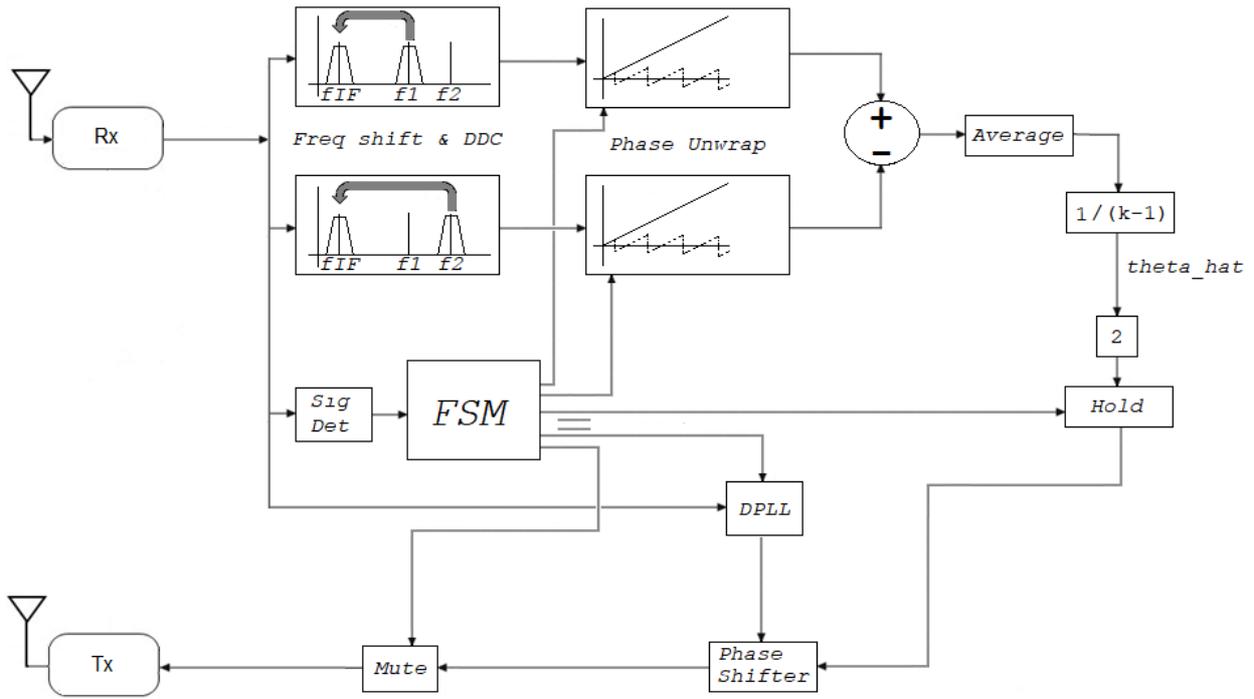


Figura B.2: Diagrama de blocos do transceptor para o método *Inversão de Fase baseado em Duplo Tom* (IFDT).

$$cal_{rf}(t) = \cos(\omega_{rf}t + \phi_{tx}), \quad (\text{B.5})$$

é recebido pela antena $RX2$ e convertido para banda base como

$$cal_{rx}(t) = \cos(\omega t + \phi_{tx} - \phi_{rx}). \quad (\text{B.6})$$

No *passo2*, um detector de fase calcula a diferença de fase entre $cal_{tx}(t)$ e $cal_{rx}(t)$ e armazena seu conjugado como

$$rtx_lo_eq = -\phi_{tx} + \phi_{rx}, \quad (\text{B.7})$$

para ser utilizado como fator constante de compensação de fase desse ponto em diante.

B.3 Resultados obtidos por meio de simulação (IFDT)

O processo descrito na seção B.1 foi testado em um ambiente simulado composto de um interrogador e dois sensores, implementados conforme o diagrama de blocos da Figura B.2.

Para efeito de comparação, durante a simulação, dois processos foram executados em paralelo sobre os mesmos sinais: um processo com o processo de correção de fase ativado e outro processo sem correção de fase.

O valor de k (relação entre f_1 e f_2) foi determinado experimentalmente. Determinou-se o valor máximo de erro de fase para um valor arbitrário de ganho de *beamforming*. Em seguida, variou-se k de 1 a 2, em busca do valor que resultasse, no máximo, nesse erro de fase, conforme ilustrado na Figura B.3, para uma exemplo com $\theta = 2,6478$ radianos e $\phi = 0,1410$ radianos.

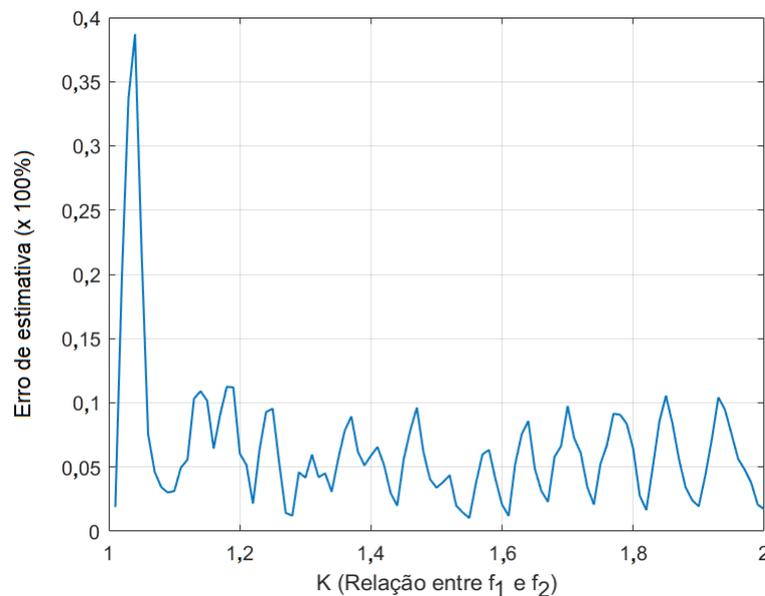


Figura B.3: Erro de estimativa de fase em função da relação entre f_1 e f_2 .

Os resultados obtidos com essa simulação estão mostrados nas Figuras B.4-B.7.

Na Figura B.4, que representa o resultado do processo sem correção de fase, está mostrada a intensidade relativa do sinal no receptor do interrogador, representada pelo traço

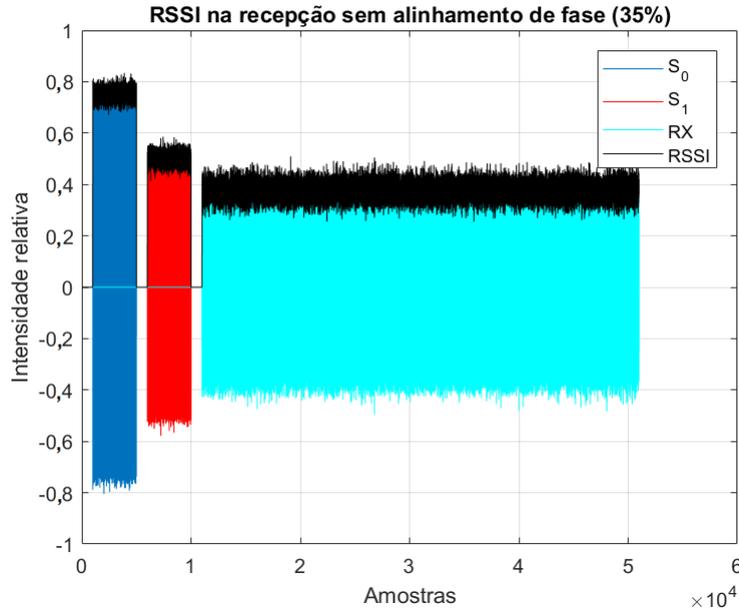


Figura B.4: Intensidade do sinal recebido no interrogador sem alinhamento de fase.

preto. O traço de cor azul representa o sinal do sensor S_0 e o traço de cor vermelha representa o sinal do sensor S_1 . Como se pode ver na representação da combinação dos dois sinais, traço de cor azul mais clara, houve uma interferência destrutiva e não houve ganho de *beamforming*. Este mesmo processo está representado na Figura B.5, que mostra mais claramente a relação de fase entre os sinais, tomando como referência o sinal piloto enviado pelo interrogador, traço preto, e o sinal resultante, tracejado preto. O resultado do processo com a correção de fase ativada, está mostrado nas Figuras B.6 e B.7. Nesse caso, nota-se que os sinais se combinam de forma coerente, ou quase-coerente, como se vê na Figura B.7, resultando em sinal cuja amplitude é maior que as amplitudes dos dois sinais quando considerados individualmente, para um ganho de *beamforming* de 84% do ganho ideal esperado, para este exemplo em particular.

B.4 Comentários

Atrasos internos à plataforma de hardware USRP, não documentados e observados empiricamente, alteram artificialmente a relação de fase entre os dois tons na saída do interrogador, impedindo a validação experimental do método IFDT na plataforma de hardware considerada (USRP).

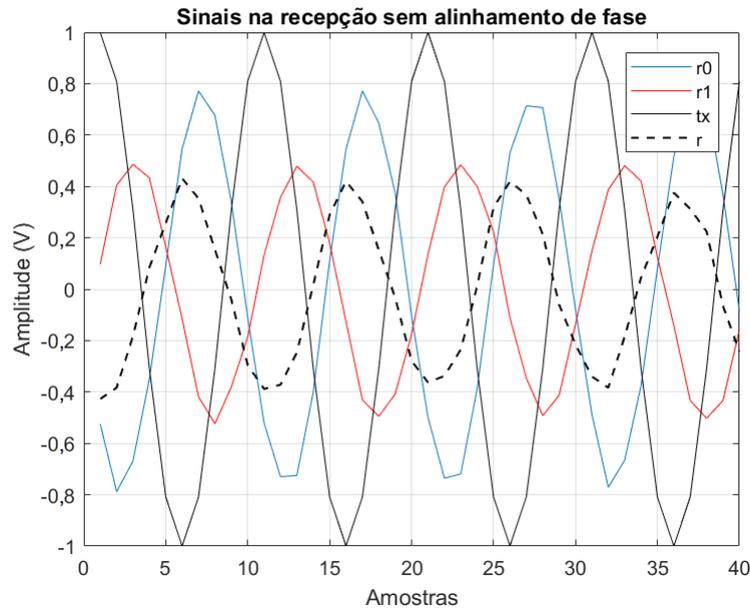


Figura B.5: Detalhe do sinal recebido no interrogador sem alinhamento de fase.

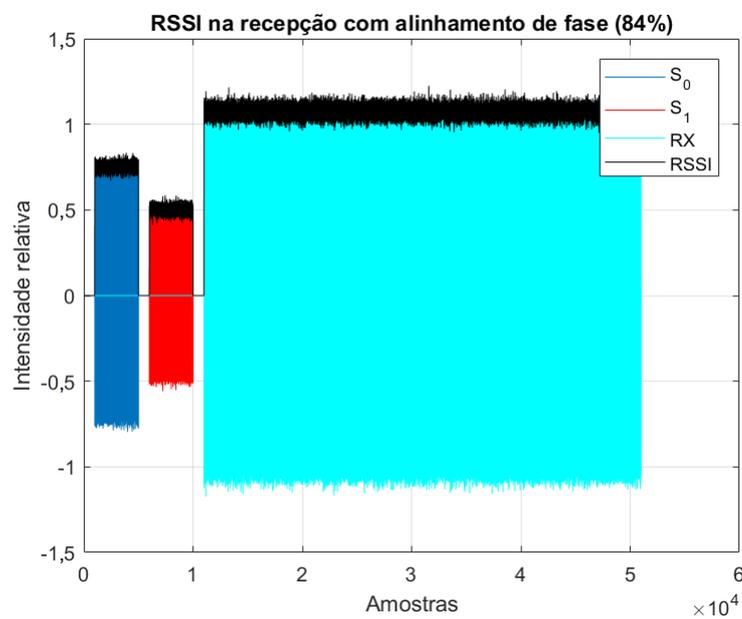


Figura B.6: Intensidade do sinal recebido no interrogador com alinhamento de fase.

Esse método, se validado em hardware apropriado, permitiria maior escalabilidade na implementação de *beamforming* distribuído, pois a latência introduzida pelo protocolo não depende do número de sensores, e as campanhas seriam compostas de ciclos com um *time-slot* para interrogação e um *time-slot* para resposta.

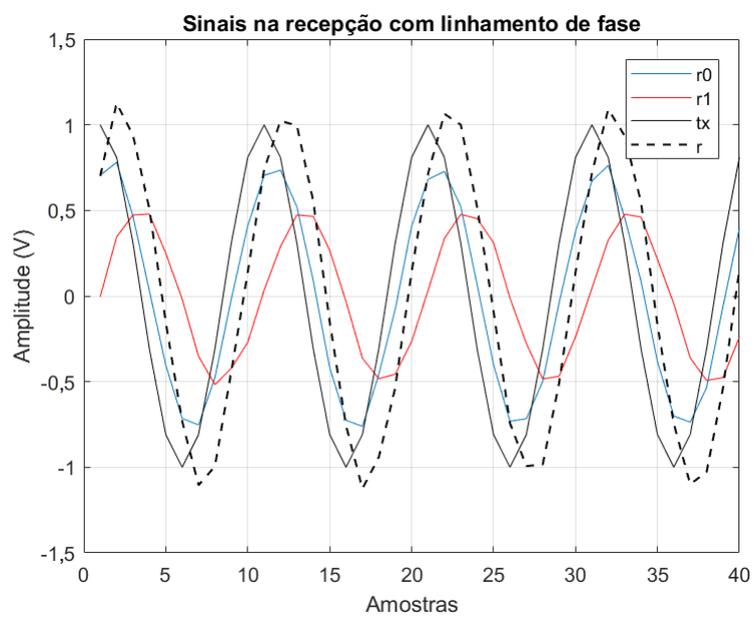


Figura B.7: Detalhe do sinal recebido no interrogador com alinhamento de fase.

Referências bibliográficas

- [1] Loureiro, A. A. F., Nogueira, J. M. S., Ruiz, L. B., Mini, R. A. F., Nakamura, E. F., Figueiredo, C. M. S., "Redes de Sensores Sem Fio", Minicurso, Simpósio Brasileiro de Redes de Computadores, Natal, RN, Março de 2003.
- [2] Kandris, D., Nakas, C., Vomvas, D., Koulouras, G., "Applications of Wireless Sensor Networks: An Up-to-Date Survey", Applied System Innovation, 2020, vol. 3, no. 14, doi: 10.3390/asi3010014.
- [3] Khan, S., Pathan, A., Alrajeh, N., "Wireless Sensor Networks: Current Status and Future Trends", CRC Press, 2016.
- [4] Ruiz, L. B., "Maná: uma arquitetura para gerenciamento de redes de sensores sem fio", Tese de Doutorado, Universidade Federal de Minas Gerais, 2003.
- [5] Karl, H., Willig, A., "Protocols and architectures for wireless sensor networks", John Wiley & Sons 2007.
- [6] Sousa, M. P., "Diversidade Cooperativa Adaptativa Aplicada a Redes de Sensores sem Fio", Dissertação de Mestrado, Universidade Federal de Campina Grande, 2009.
- [7] V. Potdar, A. Sharif and E. Chang, "Wireless Sensor Networks: A Survey", International Conference on Advanced Information Networking and Applications Workshops, Bradford, UK, 2009, pp. 636-641, doi: 10.1109/WAINA.2009.192.
- [8] Sausen, P., "Gerenciamento integrado de energia e controle de topologia em redes de sensores sem fio", Tese de Doutorado, Universidade Federal de Campina Grande, 2008.

- [9] J. Barros and S. D. Servetto, "Network information flow with correlated sources", *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 155-170, Jan. 2006, doi: 10.1109/TIT.2005.860435.
- [10] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: A survey", *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol.38, no.4, pp.393-422, March 2002.
- [11] A. Krohn, "Superimposed Radio Signals for Wireless Sensor Network", PhD Thesis, Telecooperation Office, Universitat Karlsruhe, 2006.
- [12] Glover, I., Grant, P., "Digital Communications", 3rd Edition, Person Prentice Hall, 2009.
- [13] L. Godara, "Smart Antennas", CRC Press, NY, 2004.
- [14] Comberiate, T., Zilevu, K., Hodkin, J., and Nanzer, J., "Distributed transmit beamforming on mobile platforms using high-accuracy microwave wireless positioning", *Proceedings of SPIE 9829, Radar Sensor TechnologyXX*, 2016.
- [15] R. Mudumbai, D. R. Brown III, U. Madhow and H. V. Poor, "Distributed transmit beamforming: challenges and recent progress", *IEEE Communications Magazine*, vol. 47, no. 2, pp. 102-110, February 2009, doi: 10.1109/MCOM.2009.4785387.
- [16] S. Berger and A. Wittneben, "Carrier phase synchronization of multiple distributed nodes in a wireless network", *IEEE 8th Workshop on Signal Processing Advances in Wireless Communications*, Helsinki, Finland, 2007, pp. 1-5, doi: 10.1109/SPAWC.2007.4401342.
- [17] D. R. Brown, G. B. Prince and J. A. McNeill, "A method for carrier frequency and phase synchronization of two autonomous cooperative transmitters", *IEEE 6th Workshop on Signal Processing Advances in Wireless Communications*, 2005., New York, NY, USA, 2005, pp. 260-264, doi: 10.1109/SPAWC.2005.1505912.

- [18] D. R. Brown III and H. V. Poor, "Time-Slotted Round-Trip Carrier Synchronization for Distributed Beamforming", *IEEE Transactions on Signal Processing*, vol. 56, no. 11, pp. 5630-5643, Nov. 2008, doi: 10.1109/TSP.2008.927073.
- [19] Zang, B., "Real-Time Software-Defined-Radio Implementation of Time-Slotted Carrier Synchronization for Distributed Beamforming", MSc Dissertation, WORCESTER POLYTECHNIC INSTITUTE, 2009.
- [20] Ni, Min, "An Optimized Software-Defined-Radio Implementation of Time-Slotted Carrier Synchronization for Distributed Beamforming", MSc thesis, WORCESTER POLY-TECHNIC INSTITUTE, 2011.
- [21] R. Mudumbai, G. Barriac and U. Madhow, "On the Feasibility of Distributed Beamforming in Wireless Networks", *IEEE Transactions on Wireless Communications*, vol. 6, no. 5, pp. 1754-1763, May 2007, doi: 10.1109/TWC.2007.360377.
- [22] R. Mudumbai, J. Hespanha, U. Madhow and G. Barriac, "Distributed Transmit Beamforming Using Feedback Control", *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 411-426, Jan. 2010, doi: 10.1109/TIT.2009.2034786.
- [23] I. Thibault, G. E. Corazza and L. Deambrogio, "Random, deterministic, and hybrid algorithms for distributed beamforming", *IEEE 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop*, Cagliari, Italy, 2010, pp. 221-225, doi: 10.1109/ASMS-SPSC.2010.5586901.
- [24] M. Johnson, M. Mitzenmacher and K. Ramchandran, "Distributed beamforming with binary signaling", *IEEE International Symposium on Information Theory*, Toronto, ON, Canada, 2008, pp. 890-894, doi: 10.1109/ISIT.2008.4595115.
- [25] A. F. Dana and B. Hassibi, "On the power efficiency of sensory and ad hoc wireless networks", *IEEE Transactions on Information Theory*, vol. 52, no. 7, pp. 2890-2914, July 2006, doi: 10.1109/TIT.2006.876245.

- [26] L. Dong, A. P. Petropulu and H. V. Poor, "A Cross-Layer Approach to Collaborative Beamforming for Wireless Ad Hoc Networks", *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2981-2993, July 2008, doi: 10.1109/TSP.2008.917352.
- [27] Segers, L., Tiete, J.; Braeken, A., Touhafi, A., "Ultrasonic multiple-access ranging system using spread spectrum and mems technology for indoor localization", *Sensors* 2014, 2014, 3172–3187. doi: 10.3390/s140203172.
- [28] H. Ochiai, P. Mitran, H. V. Poor and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks", *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4110-4124, Nov. 2005, doi: 10.1109/TSP.2005.857028.
- [29] K. Hardwick, D. Goeckel and D. Towsley, "Modeling distributed beamforming in wireless networks", *46th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, USA, 2008, pp. 475-482, doi: 10.1109/ALLERTON.2008.4797596.
- [30] H. Ochiai, P. Mitran, H. V. Poor and V. Tarokh, "Collaborative beamforming in ad hoc networks", *IEEE Information Theory Workshop*, San Antonio, TX, USA, 2004, pp. 396-401, doi: 10.1109/ITW.2004.1405336.
- [31] G. Barriac, R. Mudumbai and U. Madhow, "Distributed beamforming for information transfer in sensor networks", *IEEE Third International Symposium on Information Processing in Sensor Networks*, 2004. *IPSN 2004*, Berkeley, CA, USA, 2004, pp. 81-88, doi: 10.1109/IPSN.2004.238786.
- [32] Her, J., Wysoki, T. A., Wysoki, B. J., "Review of distributed beamforming", *Journal of Telecommunications and Information Technology*, vol. 1, pp. 78–88, 2011, National Institute of Telecommunications, Poland, 2011.
- [33] S. Jayaprakasam, S. K. A. Rahim and C. Y. Leow, "Distributed and Collaborative Beamforming in Wireless Sensor Networks: Classifications, Trends, and Research Directions", *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2092-2116, 2017, doi: 10.1109/COMST.2017.2720690.

- [34] H. Ochiai, P. Mitran, H. V. Poor and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks", *IEEE Transactions on Signal Processing*, vol. 53, no. 11, pp. 4110-4124, Nov. 2005, doi: 10.1109/TSP.2005.857028.
- [35] M. F. A. Ahmed and S. A. Vorobyov, "Performance characteristics of collaborative beamforming for wireless sensor networks with Gaussian distributed sensor nodes", *IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, NV, USA, 2008, pp. 3249-3252, doi: 10.1109/ICASSP.2008.4518343.
- [36] M. F. A. Ahmed and S. A. Vorobyov, "Beampattern random behavior in wireless sensor networks with Gaussian distributed sensor nodes", *IEEE Canadian Conference on Electrical and Computer Engineering*, Niagara Falls, ON, Canada, 2008, pp. 000257-000260, doi: 10.1109/CCECE.2008.4564535.
- [37] M. F. A. Ahmed and S. A. Vorobyov, "Collaborative beamforming for wireless sensor networks with Gaussian distributed sensor nodes", *IEEE Transactions on Wireless Communications*, vol. 8, no. 2, pp. 638-643, Feb. 2009, doi: 10.1109/TWC.2009.071339.
- [38] M. M. Rahman, H. E. Baidoo-Williams, R. Mudumbai and S. Dasgupta, "Fully wireless implementation of distributed beamforming on a software-defined radio platform", *ACM/IEEE 11th International Conference on Information Processing in Sensor Networks (IPSN)*, Beijing, China, 2012, pp. 305-315, doi: 10.1109/IPSN.2012.6920945.
- [39] D. R. Brown, B. Zhang, B. Svirchuk and M. Ni, "An experimental study of acoustic distributed beamforming using round-trip carrier synchronization", *IEEE International Symposium on Phased Array Systems and Technology*, Waltham, MA, USA, 2010, pp. 316-323, doi: 10.1109/ARRAY.2010.5613352.
- [40] G. Barriac, R. Mudumbai and U. Madhow, "Distributed beamforming for information transfer in sensor networks", *IEEE Third International Symposium on*

Information Processing in Sensor Networks, 2004. IPSN 2004, Berkeley, CA, USA, 2004, pp. 81-88, doi: 10.1109/IPSN.2004.238786.

[41] R. Mudumbai, J. Hespanha, U. Madhow and G. Barriac, "Distributed Transmit Beamforming Using Feedback Control", *IEEE Transactions on Information Theory*, vol. 56, no. 1, pp. 411-426, Jan. 2010, doi: 10.1109/TIT.2009.2034786.

[42] S. Sigg and M. Beigl, "Algorithms for closed-loop feedback based distributed adaptive beamforming in wireless sensor networks", *IEEE International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Melbourne, VIC, Australia, 2009, pp. 25-30, doi: 10.1109/ISSNIP.2009.5416847.

[43] S. Sigg, R. M. El Masri and M. Beigl, "Feedback-Based Closed-Loop Carrier Synchronization: A Sharp Asymptotic Bound, an Asymptotically Optimal Approach, Simulations, and Experiments", *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1605-1617, Nov. 2011, doi: 10.1109/TMC.2011.21.

[44] M. Pun, D. R. Brown and H. V. Poor, "Opportunistic collaborative beamforming with one-bit feedback", *IEEE Transactions on Wireless Communications*, vol. 8, no. 5, pp. 2629-2641, May 2009, doi: 10.1109/TWC.2009.080512.

[45] S. Song, J. S. Thompson, P. Chung and P. M. Grant, "Improving the One-Bit Feedback Algorithm for Distributed Beamforming", *IEEE Wireless Communication and Networking Conference*, Sydney, NSW, Australia, 2010, pp. 1-6, doi: 10.1109/WCNC.2010.5506562.

[46] W. Tushar and D. B. Smith, "Distributed transmit beamforming based on a 3-bit feedback system", *IEEE 11th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Marrakech, Morocco, 2010, pp. 1-5, doi: 10.1109/SPAWC.2010.5670903.

[47] G. Lim and L. J. Cimini, "Partitioned one-bit feedback for cooperative beamforming", *IEEE 44th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, USA, 2010, pp. 1-5, doi: 10.1109/CISS.2010.5464769.

- [48] F. Quitin, M. M. Ur Rahman, R. Mudumbai and U. Madhow, "Distributed beamforming with software-defined radios: Frequency synchronization and digital feedback", IEEE Global Communications Conference (GLOBECOM), Anaheim, CA, USA, 2012, pp. 4787-4792, doi: 10.1109/GLOCOM.2012.6503876.
- [49] D. R. Brown, G. B. Prince and J. A. McNeill, "A method for carrier frequency and phase synchronization of two autonomous cooperative transmitters", IEEE 6th Workshop on Signal Processing Advances in Wireless Communications, 2005., New York, NY, USA, 2005, pp. 260-264, doi: 10.1109/SPAWC.2005.1505912.
- [50] D. R. Brown III and H. V. Poor, "Time-Slotted Round-Trip Carrier Synchronization for Distributed Beamforming", IEEE Transactions on Signal Processing, vol. 56, no. 11, pp. 5630-5643, Nov. 2008, doi: 10.1109/TSP.2008.927073.
- [51] R. D. Preuss and D. R. Brown, "Retrodirective distributed transmit beamforming with two-way source synchronization", IEEE 44th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 2010, pp. 1-6, doi: 10.1109/CISS.2010.5464801.
- [52] A. Bletsas, A. Lippman and J. N. Sahalos, "Simple, zero-feedback, distributed beamforming with unsynchronized carriers", IEEE Journal on Selected Areas in Communications, vol. 28, no. 7, pp. 1046-1054, September 2010, doi: 10.1109/JSAC.2010.100909.
- [53] A. Bletsas, A. Lippman and J. N. Sahalos, "Simple, zero-feedback, collaborative beamforming for emergency radio", IEEE 6th International Symposium on Wireless Communication Systems, Siena, Italy, 2009, pp. 657-661, doi: 10.1109/ISWCS.2009.5285299.
- [54] A. Bletsas, A. Lippman, and J. N. Sahalos, "Zero-feedback, collaborative beamforming for emergency radio: Asymptotic analysis", Mobile Networks and Applications, vol. 16, no. 5, pp. 589-599, 2011, doi: 10.1007/s11036-010-0276-0.
- [55] R. Mudumbai, J. Hespanha, U. Madhow and G. Barriac, "Scalable feedback control for distributed beamforming in sensor networks", IEEE Proceedings of

International Symposium on Information Theory, 2005. ISIT 2005., Adelaide, SA, Australia, 2005, pp. 137-141, doi: 10.1109/ISIT.2005.1523309.

[56] R. Mudumbai, B. Wild, U. Madhow, and K. Ramchandran, "Distributed Beamforming using 1 Bit Feedback: from Concept to Realization", Proceedings of 44'th Allerton Conference on Communication Control and Computing, Sept. 2006.

[57] D. Rife and R. Boorstyn, "Single tone parameter estimation from discrete-time observations", IEEE Transactions on Information Theory, vol. 20, no. 5, pp. 591-598, September 1974, doi: 10.1109/TIT.1974.1055282.

[58] Suleiman, Umar Dauda, "A Review On Frequency Synchronization in Collaborative Beamforming: A Practical Approach", Penerbit Akademia Baru, 2017.

[59] Abari, Omid., Hariharan Rahul, and Dina Katabi. "One Clock to Rule Them All : A Primitive for Distributed Wireless Protocols at the Physical Layer", Computer Science and Artificial Intelligence Laboratory Technical Report, no. MITCSAIL-TR-2014-010 (2014): 1-16, Massachusetts Institute of Technology, 2014.

[60] S. Mghabghab, H. Ouassal and J. A. Nanzer, "Wireless Frequency Synchronization for Coherent Distributed Antenna Arrays", IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting, Atlanta, GA, USA, 2019, pp. 1575-1576, doi: 10.1109/APUSNCURSINRSM.2019.8889331.

[61] R. L. Schmid, T. M. Comberiate, J. E. Hodkin and J. A. Nanzer, "A Distributed RF Transmitter Using One-Way Wireless Clock Transfer", IEEE Microwave and Wireless Components Letters, vol. 27, no. 2, pp. 195-197, Feb. 2017, doi: 10.1109/LMWC.2017.2648510.

[62] GNU radio, <http://gnuradio.org>, 2020.

[63] USRP products, <http://www.ettus.com/products>, 2020.

[64] Munkyo Seo, M. Rodwell and U. Madhow, "A feedback-based distributed phased array technique and its application to 60-GHz wireless sensor network", IEEE

MTT-S International Microwave Symposium Digest, Atlanta, GA, USA, 2008, pp. 683-686, doi: 10.1109/MWSYM.2008.4632924.

[65] Y.-C. Guo, X.-W. Shi, and L. Chen, "Retrodirective Array Technology", Progress In Electromagnetics Research B, Vol. 5, 153-167, 2008, doi:10.2528/PIERB08021704.

[66] Chen, L., Y. C. Guo, X. W. Shi, and T. L. Zhang, "Overview on the phase conjugation techniques of the retrodirective array", International Journal of Antennas and Propagation, Vol. 2010, 2010, doi: 10.1155/2010/564357.

[67] C. Pon, "Retrodirective array using the heterodyne technique", IEEE Transactions on Antennas and Propagation, vol. 12, no. 2, pp. 176-180, March 1964, doi: 10.1109/TAP.1964.1138191.

[68] L. D. DiDomenico and G. M. Rebeiz, "Digital communications using self-phased arrays", IEEE Transactions on Microwave Theory and Techniques, vol. 49, no. 4, pp. 677-684, April 2001, doi: 10.1109/22.915442.

[69] Amundson I., Koutsoukos X.D. "A Survey on Localization for Mobile Wireless Sensor Networks", Fuller R., Koutsoukos X.D. (eds) Mobile Entity Localization and Tracking in GPS-less Environments. MELT 2009. Lecture Notes in Computer Science, vol 5801. Springer, Berlin, Heidelberg, doi: 10.1007/978-3-642-04385-7_16.

[70] Lanxin Qiu et al., "Multifrequency phase difference of arrival range measurement: principle, implementation, and evaluation", International Journal of Distributed Sensor Networks 11.11, p. 715307, 2015, doi: 10.1155/2015/715307.

[71] M. Pelka, C. Bollmeyer and H. Hellbrück, "Accurate radio distance estimation by phase measurements with multiple frequencies", IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), Busan, Korea (South), 2014, pp. 142-151, doi: 10.1109/IPIN.2014.7275478.

[72] N. A. C. Hassan, K. M. Yusof and S. K. S. Yusof, "Ranging Estimation Using Dual-Frequency Doppler Technique", IEEE 5th International Conference on IT Convergence and Security (ICITCS), Kuala Lumpur, Malaysia, 2015, pp. 1-5, doi: 10.1109/ICITCS.2015.7293012.

- [73] B. Sackenreuter, N. Hadaschik, M. Faßbinder and C. Mutschler, "Low-complexity PDoA-based localization", IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), Alcalá de Henares, Spain, 2016, pp. 1-6, doi: 10.1109/IPIN.2016.7743692.
- [74] Ólafsdóttir H., Ranganathan A., Capkun S. (2017) "On the Security of Carrier Phase-Based Ranging", Fischer W., Homma N. (eds) Cryptographic Hardware and Embedded Systems – CHES 2017. CHES 2017. Lecture Notes in Computer Science, vol 10529. Springer, 2017, doi: 10.1007/978-3-319-66787-4_24
- [75] H. Yan, S. Hanna, K. Balke, R. Gupta and D. Cabric, "Software Defined Radio Implementation of Carrier and Timing Synchronization for Distributed Arrays", IEEE Aerospace Conference, Big Sky, MT, USA, 2019, pp. 1-12, doi: 10.1109/AERO.2019.8742232.
- [76] J. C. Merlano-Duncan et al., "A Remote Carrier Synchronization Technique for Coherent Distributed Remote Sensing Systems", IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 1909-1922, 2021, doi: 10.1109/JSTARS.2020.3046776.
- [77] Mghabghab, S., Ellison, S., Nanzer, J., "Distributed Radio Frequency Cooperation at the Wavelength Level Using Wireless Phase Synchronization", Cornell University, 2020, <https://arxiv.org/abs/2010.10396v1>.
- [78] S. Prager, M. S. Haynes and M. Moghaddam, "Wireless Subnanosecond RF Synchronization for Distributed Ultrawideband Software-Defined Radar Networks", IEEE Transactions on Microwave Theory and Techniques, vol. 68, no. 11, pp. 4787-4804, Nov. 2020, doi: 10.1109/TMTT.2020.3014876.
- [79] X. Fan, Z. Zhang, W. Trappe, Y. Zhang, R. Howard and Z. Han, "Secret-Focus: A Practical Physical Layer Secret Communication System by Perturbing Focused Phases in Distributed Beamforming", IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, Honolulu, HI, 2018, pp. 1781-1789, doi: 10.1109/INFOCOM.2018.8486339.

- [80] S. Mohanti et al., "AirBeam: Experimental Demonstration of Distributed Beamforming by a Swarm of UAVs", IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Monterey, CA, USA, 2019, pp. 162-170, doi: 10.1109/MASS.2019.00028.
- [81] X. Fan, H. Ding, Y. Zhang, W. Trappe, Z. Han and R. Howard, "Distributed beamforming based wireless power transfer: Analysis and realization", Tsinghua Science and Technology, vol. 25, no. 6, pp. 758-775, Dec. 2020, doi: 10.26599/TST.2020.9010008.