ANÁLISE DO MOVIMENTO DE VEÍCULOS UTILIZANDO TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL

Claudio Eduardo Csura Szendrodi

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA CIVIL.

Aprovada por:

_____
Prof. Alexandre Gonçalves Evsukoff, Dr.

_____
Eng. José Vicente Calvano, D.Sc.

_____
Prof. Nelson Francisco Favilla Ebecken, D.Sc.

_____
Prof. Antonio Carneiro de Mesquita Filho, Dr.d'État.

RIO DE JANEIRO, RJ - BRASIL
ABRIL DE 2008

SZENDRODI, CLAUDIO EDUARDO CSURA

Análise do Movimento de Veículos Utilizando Técnicas de Inteligência Artificial [Rio de Janeiro] 2008

XVII, 130 p. 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia Civil, 2008)

Dissertação - Universidade Federal do Rio de Janeiro, COPPE

1. Análise de Movimento de Alvos
2. *Target Motion Analysis*
3. *Bearing-Only* - (B-O)
4. *EKELUND*
5. Pseudolinear
6. Filtros Extendidos de Kalman

I. COPPE/UFRJ II. Título (série)

Dedicatória

À minha esposa Patrícia e ao meu filhinho Ricardo,

O amor que sinto por vocês não poderia ser resumido em tão poucas linhas.

Mas mesmo assim, gostaria de falar do fundo de minha alma:

Vocês são o motivo de minhas lutas e o porto seguro de minha vida;

Vocês são o meu céu azul anil, sem nuvens ou turbulências, e onde meu vôo encontra a Paz de Deus; e

Somente vocês sabem o quanto os meus momentos de ausência, ou por estar estudando, ou por estar fazendo estes trabalho, fizeram falta...

Portanto, somente poderia dedicar estas palavras e este trabalho a vocês, com todo meu amor!

A Deus, a Jesus e a todos os meus amigos e guias espirituais,

Dedico este trabalho em agradecimento a vocês por toda a luz e o auxílio fraterno nos meus momentos de aflição, quando pensei que não iria conseguir cumprir esta missão!

Agradecimentos

Não gostaria de ser injusto com alguém por colocar uma ordem neste espaço de agradecimentos. Por isso, saibam que todos de alguma maneira, a sua maneira, me ajudaram durante minha vida profissional, e que se, de algum modo, consegui subir mais este degrau, foi porque tive a ajuda e o carinho de vocês:

A minha esposa Patrícia e ao meu filhinho Ricardo, que me deram muito amor e compreensão, em todos os momentos;

Ao meu falecido sogro, Sr. Antônio, a minha Sogra, Dna. Solange, e a Avó de minha esposa, Dna. Ilka, obrigado por tudo que vocês fizeram e fazem por mim, pela Patrícia e pelo Ricardo. Sem a ajuda e o incentivo diários de vocês, nada disso teria sido possível;

Aos meus Pais, Ildi e Jorge, por terem me dado a oportunidade da vida, e aos meus Irmãos, Débora e Rafael, por tudo que aprendemos vivendo juntos;

Aos meus Professores do Liceu de Artes e Ofícios, durante meu curso técnico, aos meus Professores do Departamento de Engenharia Eletrônica, durante minha graduação, e aos meus Professores do PEC e do PEE da COPPE, durante este mestrado, por todos os ensinamentos que consegui adquirir;

Aos amigos do LASPI (GEMD) do Departamento de Engenharia Eletrônica da UFRJ, em especial ao amigo Prof. Casé, orientador na graduação, que me permitiu utilizar as instalações e os computadores do laboratório durante o desenvolvimento desta tese, quando os recursos disponíveis em outros locais eram insuficientes;

À Marinha do Brasil, representada pela Diretoria-Geral do Material da Marinha, pela Diretoria de Sistemas de Armas da Marinha e pelo Centro de Eletrônica da Marinha, por terem me permitido realizar este curso de carreira e utilizar seus recursos materiais no desenvolvimento deste trabalho.

Aos Senhores Diretores, meus Comandantes e ex-Comandantes, da DSAM e do CETM, Vice-Almirante(RM1) Uzeda, Vice-Almirante Fernandes e Contra-Almirante Marcos José, pela oportunidade do curso e pelos conselhos ao longo do mesmo;

Aos Meus Vice-Diretores, da DSAM e do CETM, CMG(EN) Gomes Braga, CMG Péricles e CMG José Augusto, que muito me ajudaram antes e durante o curso;

Aos Meus Superintendentes e Chefes de Departamento, da DSAM e do CETM, por acreditarem no meu trabalho, incentivando e ajudando de todas as formas possíveis para que conseguisse chegar até aqui. Ao CMG(RM1-EN) Fernando que indicou desde o início o rumo a ser seguido para este trabalho, ou seja, o tema da tese;

Aos meus Amigos da Marinha, CT(EN) Savioli, CT(EN) Luciano, agradeço o companheirismo, a ajuda durante o trabalho, a lealdade, e sobre tudo a amizade;

Aos Amigos de Praça D'Armas, da DSAM e do CETM, e do PEC da COPPE;

Ao meu Amigo, Orientador, Professor, Doutor, Capitão-de-Corveta(EN) José Vicente Calvano, pela dedicação, pela orientação sempre segura, por confiar no meu trabalho e principalmente pela amizade. Se não fosse por seu empenho pessoal, este mestrado não teria sido possível desde o início;

Ao Professor, Doutor, Nelson Francisco Favilla Ebecken, quando aceitou minha inscrição e me concedeu a possibilidade deste curso, que poderia não acontecer por outros fatores, agradeço sinceramente por tudo que aprendi com o senhor!;

Ao meu Amigo e Orientador, Professor, Doutor, Alexandre Gonçalves Evsukoff, pela orientação sempre segura, por acreditar em mim desde o início, me ajudando e incentivando, em todos os momentos.

Finalmente, agradeço ao meu Bom Deus e ao meu Amigo Jesus, que me protegem nos momentos mais difíceis, me orientam nos momentos certos e que acalmam minha alma quando os caminhos se tornam sinuosos.

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

ANÁLISE DE MOVIMENTO DE VEÍCULOS UTILIZANDO TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL

Claudio Eduardo Csura Szendrodi

Abril/2008

Orientadores: Prof. Alexandre Gonçalves Evsukoff
Eng. José Vicente Calvano

Programa: Engenharia Civil

Esta dissertação trata da análise do movimento de veículos utilizando técnicas de inteligência artificial. A proposta da tese e o seu desenvolvimento consistem dos seguintes tópicos: apresentação de um estudo teórico sobre técnicas e algoritmos utilizados em problemas tradicionalmente conhecidos como *Target Motion Analysis* (TMA); proposição e desenvolvimento de um simulador de movimentos de contatos; avaliação das técnicas e dos algoritmos inicialmente estudados; e, por fim, foi proposto, implementado e avaliado um novo método que utiliza conceitos e técnicas de inteligência artificial, a fim de estimar e predizer a posição e a velocidade de veículos navais, de superfície e submarinos.

Os resultados obtidos pelo novo método apresentam precisão, igual ou superior ao estado da arte.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

ANALYSIS OF MOVEMENT OF VEHICLES USING TECHNIQUES OF ARTIFICIAL INTELLIGENCE

Claudio Eduardo Csura Szendrodi

April/2008

Advisors: Prof. Alexandre Gonçalves Evsukoff
Eng. José Vicente Calvano

Department: Civil Engineering

This dissertation treats of the analysis of the movement of vehicles using artificial intelligence techniques. The proposal of the thesis and her development consist of the following topics: presentation of a theoretical study on techniques and algorithms used in problems traditionally known like *Target Motion Analysis* (TMA); proposition and development of a simulator of movements of targets; evaluation of the techniques and of the algorithms initially studied; and, finally, it was proposed, implemented and appraised a new method that uses concepts and artificial intelligence techniques, in order to esteem and to predict the position and the speed of naval vehicles, of surface and submarines.

The results obtained by the new method present precision, equal or superior to the state of the art.

Índice

1. Introdução..... 1

1.1. Histórico: O Brasil e a Amazônia Azul 1

1.2. A Política Externa Brasileira: Pedido de aumento da Plataforma Continental. 1

1.3. O papel da Marinha do Brasil na Proteção da Amazônia Azul 3

1.4. Compilação de um Teatro de Operações 4

1.5. Pesquisa e Organização da Tese 5

2. Introdução Teórica, Definições, Fundamentos e Algoritmos Existentes. 6

2.1. Introdução Teórica e Definições 6

2.2. Fundamentos do problema de Análise de Movimento de Alvos (TMA) 8

2.3. Algoritmos Empregados na Estimação da Posição e da Velocidade do Contato 10

2.3.1. Método da Distância EKELUND 10

2.3.2. Algoritmos baseados em Sistemas Fuzzy 11

2.3.3. Algoritmos Estimadores Pseudolineares 11

2.3.4. Filtros de Kalman e Filtros de Kalman Estendidos 12

2.4. Transformadas Wavelets aplicadas à redução de ruído (denoising) 13

2.4.1. Transformadas *Wavelets* – o que são? 14

2.4.2. Famílias de *Wavelets* mais comumente utilizadas: 16

3. Revisão Bibliográfica, Comentários e Estado da Arte. 19

3.1. Trabalhos Pesquisados 19

3.1.1. Um Sistema de Esclarecimento para um Navio Patrulha com Deslocamento de 1300 Toneladas 19

3.1.2. Fuzzy Logic-Based Information Processing in Submarine Combat Systems 22

3.1.3. A Fuzzy Logic-Based Intelligent Controller for Contact Management Data Integration 25

3.1.4. Fuzzifying a Target Motion Analysis Model Using Fril and Mathematica 31

3.1.5. An Asymptotically Unbiased Estimator for Bearings-Only and Doppler-Bearing Target Motion Analysis 32

3.1.6. Recursive Bayesian Inference On Stochastic Differential Equations 34

3.2. Comentários adicionais sobre a revisão bibliográfica e o estado da arte 35

4. Proposição da Tese 38

4.1. Objetivo Principal 38

4.2. Pressupostos Empregados na Proposta de Solução do Problema 40

4.2.1. Simulador do Movimento do Contato e do Navio Observador 40

4.2.2. Pressupostos do Movimento do Contato e do Navio Observador 41

4.2.3. Métrica de erro adotada – Raiz quadrada do erro médio quadrático (RMSE) 42

4.2.4. Emprego dos Algoritmos de Estimação Pseudolinear 43

4.3. Resumo do Algoritmo PL Modificado Implementado para o Método Estimador Pseudolinear Modificado 53

4.4. Emprego do Algoritmo de Estimação EKELUND para comparação com os algoritmos PL Original e PL Modificado 56

4.5. Implementação da redução de ruído (denoising) com Wavelets 57

4.5.1. Função *wden* (*wavelet denoising*) da *toolbox* de *wavelets* do *MATLAB* 57

4.5.2. Considerações para este trabalho no uso da função *wden* 58

5. Padrões de Teste (*Benchmark*) 59

6. Apresentação e Análise de Resultados 64

6.1. Simulação do Cenário "A" utilizando o Algoritmo PL Original 69

6.2. Simulação dos Cenários "B" utilizando os Algoritmos EKELUND, PL Original e PL Modificado. 72

6.2.1. Simulação do Cenário "B1" utilizando o Algoritmo PL Original com intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente. 72

6.2.2. Simulação do Cenário "B1" utilizando o Algoritmo PL Modificado com intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente. 75

6.2.3. Simulação dos Cenários "B1" a "B5" - intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente..... 78

6.2.4. Simulação dos Cenários "B1" a "B5" - intervalo de tempo e tempo total de amostragem (janela de tempo) de 2s e 240s, respectivamente..... 83

6.3. Simulação do Cenário "A" utilizando os Algoritmo PL Modificado 88

6.4. Simulação dos Cenários "C", "D", e "E" utilizando os Algoritmos EKELUND, PL Original e PL Modificado. 91

6.4.1. Simulação do Cenário "C". 91

6.4.2. Simulação do Cenário "D". 96

6.4.3. Simulação do Cenário "E". 101

6.5. Wavelets Denoising empregada na redução de ruído em problemas de análise de movimento de alvos - TMA. 106

6.5.1. *Wavelet Denoising* utilizando o Algoritmo PL Modificado no Cenário "A"..... 107

6.5.2. Resultados do emprego de *Wavelets Denoising* para os cenários "B1" e "B5" utilizando os Algoritmos EKELUND e PL Modificado..... 111

6.6. Tempo de Execução dos Algoritmos EKELUND, PL Original e PL Modificado em um mesmo computador. 116

7. Conclusões 122

8. Sugestões para trabalhos futuros 124

9. Referências Bibliográficas 125

APÊNDICE I. Artigos 130

Índice de Figuras

Figura 2.1: Cenário Tático de navios – Marcações, Distâncias, Rumos e Velocidades.. 7

Figura 2.2: Movimento típico de navios durante um acompanhamento e estimação..... 8

Figura 2.3: Decomposição de um sinal em quatro níveis - análise de *wavelet* 14

Figura 2.4: Representação de uma *wavelet* de *Haar* de análise e sua função de escalamento *Haar* de análise 16

Figura 2.5: *Wavelet* de *Daubechies* de nível 16 (db16) de análise e sua respectiva função de escalamento..... 16

Figura 2.6: *Wavelet* da família *Symmlet* de nível 8 (sym8) de análise e sua função de escalamento 17

Figura 3.1: Rumos e Velocidades do Navio Observador nas simulações da dissertação em análise. 20

Figura 3.2: Resultados da possível posição do contato após o processamento com o algoritmo TMA EKELUND. 21

Figura 3.3: Possíveis resultados para uma problema de TMA para o paper em discussão [2]. 23

Figura 3.4: FIS – *Fuzzy Inference System* (ou Sistema Fuzzy de Inferência) implementado no artigo [2]. 24

Figura 3.5: Posições Verdadeiras, Estimativas e Regiões de Incerteza do Contato e do Navio Observador [2]. 24

Figura 3.6: *Intelligent Process Controller* (IPC) – Controlador Inteligente de Processos - proposto em [3]. 26

Figura 3.7: *Fuzzy Inference System* (FIS) – Sistema Fuzzy de Inferência –- proposto em [3] 26

Figura 3.8: Fuzzificação de algumas entradas e saídas utilizadas no problema do trabalho [3] em análise. 27

Figura 3.9: Outras fuzzificação de entradas e de saídas utilizadas no problema do trabalho [3] em análise. 27

Figura 3.10: Familias de possíveis soluções de posições do contato e localização do contato e do navio observador [3]. 28

Figura 3.11: Velocidade do contato pode ser caracterizada por uma função de pertinência Fuzzy – artigo [3]. 29

Figura 3.12: Resultado após o processamento Fuzzy utilizando os pesos determinados – artigo [3]. 29

Figura 3.13: Resultado da caracterização dafuzzificação das incertezas de entrada pelo FIS – artigo [3]. 30

Figura 3.14: Resultado da saída de velocidade após a defuzzyficação – artigo [3]. 30

Figura 3.14: Resultados de possíveis posições do contato após o processamento com o algoritmo Manual Inteligente que foi implementado no trabalho em análise em *FRIL* –artigo [4]. 31

Figura 3.15: Exemplo de estimação da posição do contato usando filtros EKF, UKF e *Smoothers* – Tese [5]. 35

Figura 4.1: Situação típica de cinemática de movimento de navios. 40

Figura 4.2: Posições, Marcações, Distâncias, Rumos e Velocidades dos Navios - Método Pseudolinear Original. 44

Figura 6.1: Tela de interface com o usuário do Sistema de Acompanhamento de Navios desenvolvido utilizando o pacote *MATLAB* 66

Figura 6.2: Resultado da Simulação com Cenário “A” utilizando o Algoritmo PL Original 69

Figura 6.3: Ampliações do Resultado da Simulação com Cenário “A” utilizando o Algoritmo PL Original 70
Figura 6.4: Resultado da Simulação para o cenário “B1” utilizando os Algoritmos EKELUND e PL Original..... 73
Figura 6.5: Resultado da Simulação para o cenário “B1” utilizando os Algoritmos EKELUND e PL Modificado 76
Figura 6.6: Resultado da Simulação para o cenário “B4” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 79
Figura 6.7: Resultado da Simulação para o cenário “B4” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 80
Figura 6.8: Resultado da Simulação para o cenário “B4” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 84
Figura 6.9: Resultado da Simulação para o cenário “B4” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 85
Figura 6.10: Resultado da Simulação com Cenário “A” utilizando o Algoritmo PL Modificado 88
Figura 6.11: Ampliações do Resultado da Simulação com Cenário “A” utilizando o Algoritmo PL Modificado 89
Figura 6.12: Resultado da Simulação para o cenário “C1” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 92
Figura 6.13: Resultado da Simulação para o cenário “C5” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 93
Figura 6.14: Resultado da Simulação para o cenário “D3” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 97

Figura 6.15: Resultado da Simulação para o cenário “D5” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 98

Figura 6.16: Resultado da Simulação para o cenário “E1” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 102

Figura 6.17: Resultado da Simulação para o cenário “E3” utilizando os Algoritmos EKELUND, PL Original e PL Modificado 103

Figura 6.18: : Resultado dos testes com *Wavelets Denoising* para o cenário “B1”, nível de ruído adicionado de 0,5%, utilizando os Algoritmos EKELUND e PL Modificado. 112

Figura 6.19: Resultado dos testes com *Wavelets Denoising* para o cenário “B1”, nível de ruído adicionado de 1,0%, utilizando os Algoritmos EKELUND e PL Modificado. 113

Índice de Tabelas

Tabela 4.1: Extrato resumido do Algoritmo Estimador PL Original para um problema de BO-TMA. 53

Tabela 4.2: Resumo do Algoritmo Implementado para o Método Estimador Pseudolinear Modificado. 56

Tabela 5.1: Relação dos cenários utilizados simulações. 61

Tabela 5.2: Resumo das condições iniciais para o cenário "B" 62

Tabela 5.3: Resumo das condições iniciais para o cenário "C" 62

Tabela 5.4: Resumo das condições iniciais para o cenário "D" 63

Tabela 5.5: Resumo das condições iniciais para o cenário "E" 63

Tabela 6.1: Resumo dos testes efetuados para o Cenário "A" utilizando o Algoritmo PL Original. 71

Tabela 6.2: Resumo das Condições Iniciais para o Cenário "A" utilizando o Algoritmo PL Original. 71

Tabela 6.3: Resumo das condições iniciais para o simulador MRU gerar o cenário "B1". 72

Tabela 6.4: Resumo dos testes efetuados para o cenário "B1" utilizando os Algoritmos EKELUND e PL Original 74

Tabela 6.5: Resumo das condições iniciais para o simulador MRU gerar o cenário "B1". 75

Tabela 6.6: Resumo dos testes efetuados para o cenário "B1" utilizando os Algoritmos EKELUND e PL Modificado. 77

Tabela 6.7: Resumo das condições iniciais para o simulador MRU gerar os cenários "B1" a "B5" - Algoritmos EKELUND, PL Original e PL Modificado. 78

Tabela 6.8: Resumo dos testes efetuados para os cenários “B” utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 12s e 240s, respectivamente. 82

Tabela 6.9: Resumo das condições iniciais para o simulador MRU gerar os cenários “B1” a “B5” - Algoritmos EKELUND, PL Original e PL Modificado. 83

Tabela 6.10: Resumo dos testes efetuados para os cenários “B1” a “B5” utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 87

Tabela 6.11: Resumo dos testes efetuados para o Cenário “A” utilizando o Algoritmo PL Modificado. 90

Tabela 6.12: Resumo das Condições Iniciais para o Cenário “A” utilizando o Algoritmo PL Modificado. 90

Tabela 6.13: Resumo das condições iniciais para o simulador MRU gerar os cenários “C1” a “C5” - Algoritmos EKELUND, PL Original e PL Modificado. 91

Tabela 6.14: Resumo dos testes efetuados para os cenários “C1” a “C5” utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 95

Tabela 6.15: Resumo das condições iniciais para o simulador MRU gerar os cenários “D1” a “D5” - Algoritmos EKELUND, PL Original e PL Modificado. 96

Tabela 6.16: Resumo dos testes efetuados para os cenários “D1” a “D5” utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 100

Tabela 6.17: Resumo das condições iniciais para o simulador MRU gerar os cenários “E1” a “E5” - Algoritmos EKELUND, PL Original e PL Modificado. 101

Tabela 6.18: Resumo dos testes efetuados para os cenários “E1” a “E5” utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 105

Tabela 6.19: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'sqtwolog'* para o Cenário “A” e o Algoritmo PL Modificado. 107

Tabela 6.20: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'rigrsure'* para o Cenário “A” e o Algoritmo PL Modificado. 108

Tabela 6.21: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'heursure'* para o Cenário “A” e o Algoritmo PL Modificado. 109

Tabela 6.21: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'minimaxi'* para o Cenário “A” e o Algoritmo PL Modificado. 110

Tabela 6.23: Resumo das condições iniciais para o simulador MRU gerar os cenários “B1” a “B5” para testes com *Wavelets Denoising* – Algoritmos EKELUND e PL Modificado. 111

Tabela 6.24: Resultados do emprego de *Wavelets Denoising* para o cenário “B1”, nível de ruído adicionado de 1,0%, utilizando os Algoritmos EKELUND e PL Modificado. 115

Tabela 6.25: Tempo de execução utilizando Cenário “A” utilizando os Algoritmos PL Original e PL Modificado. 116

Tabela 6.26: Tempo de execução utilizando Cenário “B” utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 12s e 240s, respectivamente. 117

Tabela 6.27: Tempo de execução utilizando Cenário “B” utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 118

Tabela 6.28: Tempo de execução utilizando Cenário “C” utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 119
Tabela 6.29: Tempo de execução utilizando Cenário “D” utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 120
Tabela 6.30: Tempo de execução utilizando Cenário “E” utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente. 121

Convenções Tipográficas

As convenções seguintes são utilizadas nesta tese para identificar certos tipos de informação:

CONVENÇÃO	DESCRIÇÃO
MAIÚSCULAS	siglas ou acrônimos
itálico	palavras ou expressões em língua inglesa ou palavras em português com significado ligeiramente diferente do habitual já definidas anteriormente no texto
sublinhado	variáveis
CAIXA ALTA	termos ou expressões sob definição

Notações Gerais:

a, b, c, x, y, t, α, β	grandeza escalares
a, b, c, x, t, α, β, θ	vetores
A, B, X, Y, F, R, W	Matrizes

Notações Convencionadas:

$\frac{dg(t)}{dt} = \dot{g}$	Derivada da função g(t) com relação a t
$\frac{\partial g(t)}{\partial t}$	Derivada parcial de g(t) com relação a t
*diag***A**	Diagonal da Matriz A
E[x]	Valor esperado de x
E[x/y]	Valor esperado de x dado y
$\mathbf{r}_{\beta,i} = \mathbf{r}(\beta_i)$	Vetor **r** em função das marcações β_i

Nomenclatura

TMA – *Target Motion Analysis*

EKF – *Extended Kalman Filter*

BO – *Bearing-Only*

PL – *Pseudolinear Estimator*

MB – Marinha do Brasil

IA – *Inteligência Artificial*

AG – *Algoritmo Genético*

PS – *Particle Swarm*

MRU - Movimento Retilíneo Uniforme

MRUV - Movimento Retilíneo Uniformemente Variado

NV – Norte Verdadeiro

Nmg - Norte Magnético

NM – *Nautical Miles*, ou milhas náuticas

CLS – *Constrained Least Squares*

1. Introdução

1.1. Histórico: O Brasil e a Amazônia Azul

O Brasil é um país que tem vocação marítima [30] desde seu descobrimento, pois foi pelo mar que chegaram os navegantes portugueses no ano de 1500.

Por sua vez, o mesmo mar que despertou e continua a motivar interesses comerciais, sempre foi usado como importante via de comunicação e transporte com os outros continentes até o início do século XX. Talvez esse interesse tenha diminuído um pouco, principalmente com relação à maioria da população brasileira.

No entanto, a crise do petróleo durante a primeira metade da década 70, mostrou que o país deveria procurar fontes alternativas de energia, ao uso dos combustíveis de origem fóssil, sobretudo do petróleo. Neste contexto surgiram algumas iniciativas: o Pró-álcool, o desenvolvimento de pesquisas e o desenvolvimento de novas tecnologias aplicadas à exploração de petróleo em águas profundas pela PETROBRAS, entre outras.

A descoberta de grandes reservas petrolíferas nos campos de Garoupa e de Namorado, na Bacia de Campos, no final da década de 70, e a exploração comercial dos mesmos em águas profundas, a partir da década de 90, permitiu ao país enfrentar a crise de energia, chegando aos dias atual, praticamente auto-suficiente [31,32]. Diversas reservas de petróleo e de gás ainda estão sendo encontradas em alto-mar, o que motivou o interesse do país pela soberania do mar e de suas riquezas, além de sua plataforma continental.

1.2. A Política Externa Brasileira: Pedido de aumento da Plataforma Continental.

No passado [30], as nações que possuem alguma fronteira territorial terminando no mar, i.e. os estados costeiros, aceitavam a existência do denominado mar territorial com três milhas náuticas de largura a contar da linha da costa.

A Organização das Nações Unidas (ONU) passou a discutir, no final da década de 50, a elaboração da Convenção das Nações Unidas sobre os Direitos do Mar. O resultado das informações advindas das pesquisas cientificas apontavam para o aumento

dos recursos naturais existentes no mar e sob o fundo do mesmo, mostrando a necessidade da existência de um novo ordenamento jurídico sobre o mar, a fim de permitir resolver disputas e beligerâncias advindas de interesses conflitantes.

No início da década dos 70, alguns países, como o Brasil, adotaram através de legislação o seu mar territorial com 200 milhas, antes de a Convenção entrar em vigor.

A ONU adotou formalmente a Convenção das Nações Unidas sobre os Direitos do Mar no ano de 1982, com os limites que vinham sendo discutidos e adotados até então. A mesma foi ratificada pelo Brasil, mas nem todos os países o fizeram ainda, como os Estados Unidos da América, por possuírem interesses pelo mundo.

A Convenção permitiu que os estados costeiros pudessem apresentar, à Comissão de Limites da Plataforma Continental da ONU (CLPC), os seus pleitos sobre o prolongamento da Plataforma Continental (PC), que excedesse as 200 milhas da sua ZEE, até um limite de 350 milhas, a partir da linha da costa. Nesse prolongamento, o estado costeiro tem direito à exploração e explotação dos recursos do solo e subsolo marinhos, mas não dos recursos vivos da camada líquida.

Até o presente momento, apenas dois países exerceram esse direito:

O primeiro foi a Rússia, que não teve ainda os seus pleitos atendidos. Um fato recente [35] exemplifica este interesse: A Rússia enviou uma missão tripulada e fincou sua bandeira a 4.200 metros de profundidade no mar do Pólo Norte, sob a alegação de que a cordilheira submarina faz parte da Sibéria. Esta região é rica em petróleo, gás e metais, sendo também disputada por Estados Unidos, Noruega, Canadá e Dinamarca.

O segundo país a exercer este direito foi o Brasil, através de ações que duraram cerca de dez anos. Navios de pesquisa da Diretoria de Hidrografia e Navegação (DHN) da MB coletaram dados por cerca de 150.000 km. O Brasil apresentou, em setembro de 2004, a sua proposta àquela comissão. Uma decisão parcial foi emitida em marco de 2007 pela CLPC [33,34], recomendando que o governo brasileiro apresente uma nova proposta que permitirá que o Brasil incorpore entre 773.000 km^2 e 963.000 km^2.

O país poderia ficar com um total de 4.400.000 km^2. Esta área corresponde à metade do território terrestre nacional, sendo uma “nova Amazônia”. É o que a MB vem chamando de “Amazônia Azul” na tentativa de mostrar a importância, não só estratégica, mas também econômica, do mar fronteiriço com o nosso litoral.

Pelo mar circulam cerca de 95% do nosso comércio exterior, cujo resultado acumulado em 2007 foi de US$ 160,649 bilhões, em exportações, e US$ 120,621 bilhões, em importações, com saldo positivo de US$ 40,028 bilhões. [36]

Das plataformas localizadas na ZEE extraímos aproximadamente 80% da nossa produção de petróleo, cerca de 2 milhões de barris/dia o que, a preços atuais, é da ordem de US$ 2 bilhões por mês. Podem-se ainda citar outros setores com potencial econômico: o pesqueiro, a cabotagem, o turismo, os esportes náuticos, e futuramente a exploração dos nódulos polimetálicos existentes no leito do mar.

O Brasil também tem o compromisso de realizar operações de busca e salvamento em uma extensa área marítima, que avança pelo Oceano Atlântico, ultrapassando os limites da Amazônia Azul.

1.3. O papel da Marinha do Brasil na Proteção da Amazônia Azul

Os limites marítimos brasileiros são linhas imaginárias. A existência de navios da MB realizando patrulhamento ou controle do tráfego marítimo é o que as define.

Existe uma extensa área, cerca de 4,5 milhões de quilômetros quadrados, a ser vigiada. Se a ação não for contínua e permanente, diversos tipos de ilícitos como pirataria, contrabando, exploração de recursos naturais podem ocorrer. As ações de patrulha que são realizadas pela MB têm objetivo preventivo, evitando que ocorram ilícitos nas águas jurisdicionais brasileiras.

O transporte marítimo representa como vimos o mais importante meio de movimentação de mercadorias para o nosso comércio exterior. O país é de tal modo dependente do tráfego marítimo que ele se constitui em uma de suas atividades de caráter mais sensível. O motivo é determinado pela dependência da indústria nacional que emprega uma grande quantidade de insumos importados.

Na falta do petróleo, de modo similar como ocorreu na década de 70, uma crise energética e de disponibilidade de insumos derivados desta matriz de energia, como os polímeros e os plásticos, podem paralisar o país.

Um modelo de Sistema de Vigilância para Amazônia Azul é tão complexo quanto o Sistema de Vigilância de Amazônia (SIVAM) implementado no continente. A MB necessita estar adequadamente aparelhada, tanto ao nível dos meios operativos, quanto ao nível de qualificação de seu pessoal, quer este seja técnico ou operativo.

Por isso o permanente interesse da MB em qualificar seu pessoal nos centros de ensino e pesquisa, nacionais e internacionais, de comprovada relevância acadêmica. Somente o domínio das tecnologias envolvidas pode permitir ao Estado Brasileiro manter uma atitude pró-ativa na defesa dos seus interesses estratégicos.

1.4. Compilação de um Teatro de Operações

Um navio realizando operações no mar necessita de diversas informações, atualizadas em tempo real, sobre o meio, acima e/ou abaixo d'água, no qual está inserido taticamente. Deste modo, não somente os supostos amigos ou inimigos devem ser monitorados, mas também suas próprias condições e parâmetros internos, bem como as condições do meio em si.

O modo como ele pode realizar este monitoramento depende essencialmente da capacidade dos seus sensores de "enxergar" muito além da linha do horizonte. Os mesmos devem adquirir o máximo de informações possíveis sobre este ambiente e convertê-las em sinais elétricos analógico e/ou digitais.

Normalmente as informações relevantes no que concerne aos navios e aeronaves ao seu redor são as posições estimadas dos mesmos.

O processamento dos sinais de sonar e de radar permite que sejam analisadas detalhadamente essas informações e também que sejam extraídos outros dados relevantes sobre os contatos apresentados por estes sensores.

O interesse deste trabalho está na extração de informações cinemáticas dos contatos acompanhados, a partir somente das marcações relativas para o contato e da posição do navio observador ao longo de um intervalo de tempo. Trataremos prioritariamente, neste trabalho, do acompanhamento de contatos submarinos e de superfície, mas os algoritmos podem ser utilizados também para contatos aéreos.

Vale ressaltar que a importância do acompanhamento de contatos esta ligada não somente as questões de ordem militar, ou estratégica, mas também à segurança da navegação, uma vez que mesmo os navios mercantes e os grandes petroleiros necessitam estimar a posição de outras embarcações de modo a evitar desastres que possam causar perdas de vidas humanas, ou grandes desastres ecológicos.

Portanto, a determinação dos parâmetros da cinemática dos contatos e a predição de sua posição nos instantes seguintes a partir dos dados anteriormente amostrados pelos sensores de bordo são informações relevantes não somente no auxílio à navegação, como também desempenham importante papel na vigilância, na compilação e controle do quadro tático, e, por conseguinte, no correto controle e uso dos sistemas navais.

1.5. Pesquisa e Organização da Tese

O trabalho de pesquisa de Tese foi iniciado com uma Revisão Bibliográfica, que, compila os principais artigos publicados sobre o tema proposto, nos últimos quinze anos.

Os artigos compilados durante a revisão, mostram que as seguintes técnicas têm sido avaliadas na comunidade científica: Lógica *Fuzzy*, Sistemas *Fuzzy* Recursivos, Séries Temporais, Algoritmos Genéticos, Redes Neuronais, Sistemas Híbridos (e.g. *Neuro-Fuzzy* ou *AG-Fuzzy*), Transformadas de Fourier e *Wavelets*. Por este motivo, tais técnicas foram estudadas para um possível emprego na presente Tese.

No capítulo dois, são apresentadas: a introdução teórica; as definições usadas para sua descrição; e bem como sua formulação. Os princípios de funcionamento dos métodos pesquisados são também apresentados neste capítulo.

A revisão bibliográfica sobre o assunto, alguns comentários adicionais sobre a revisão bibliográfica e o estado da arte estão desenvolvidos no capítulo três.

O capítulo quatro apresenta a proposição da tese. No mesmo, encontra-se a descrição do método que foi implementado nesta dissertação, a partir de um trabalho publicado em 2006 [5] que implementa algoritmos através de Estimadores Pseudolineares (PL) utilizando minimização forçada de mínimos quadrados, em um problema de TMA.

O capítulo cinco apresenta os padrões de teste (*benchmark*) utilizados através dos padrões cinemáticos de movimento de navios e contatos que foram empregados em trabalhos anteriores e também nesta dissertação.

O capítulo seis encontra-se a apresentação e a análise dos resultados alcançados.

Os capítulos sete e oito apresentam, respectivamente, as conclusões e os comentários adicionais, bem como sugestões para trabalhos futuros. Por fim, são apresentadas as referências bibliográficas utilizadas neste trabalho.

2. Introdução Teórica, Definições, Fundamentos e Algoritmos Existentes.

Neste capítulo estão reunidas, de modo sucinto, as considerações teóricas inicias, os conceitos, as definições, os fundamentos e os termos comumente associados às técnicas que empregam estimadores na busca e otimização da solução de posição de contatos.

Os principais algoritmos e técnicas existentes são citados nesta introdução.

2.1. Introdução Teórica e Definições

No presente trabalho, será designado como "contato" todo e qualquer objeto passível de se controlar por meios próprios, possuidor de algum intento, mesmo que somente o de observação.

Comumente é utilizado na literatura o termo "alvo", do inglês *target*, que designa o "contato" quando o mesmo possui algum tipo de interesse contrário ao do navio observador. No entanto, mesmo que os autores dos trabalhos estudados ao longo desta tese utilizem "alvo" para ser referir ao objeto que está sendo observado, o mesmo será aqui chamado de "contato", uma vez que o trabalho ora apresentado é de cunho acadêmico.

O navio "observador" é a plataforma que possui os sensores passivos ou ativos que realizam o monitoramento dos contatos. Normalmente será o navio onde se encontra o observador nas coordenadas iniciais nulas.

O termo "pernada" é empregado para nomear a trajetória que um navio qualquer, sendo normalmente o navio observador, descreve na superfície da água ou dentro da mesma, caso seja um submarino, quando seu rumo e sua velocidade não sofrem variações significativas.

Define-se o termo "corrida" como sendo o conjunto de movimentos que o navio observador efetua sob ou sobre o meio líquido, sendo formado de uma ou mais "Pernadas", dentro de um intervalo de tempo de observação.

Será definido como "acompanhamento de contatos" o processo de determinação dos parâmetros de interesse daquele contato, que normalmente são os parâmetros cinemáticos estimados de posição, rumo, velocidade e aceleração.

Na figura 2.1, encontra-se representado um exemplo de cenário tático de navios, com a representação das principais variáveis envolvidas.

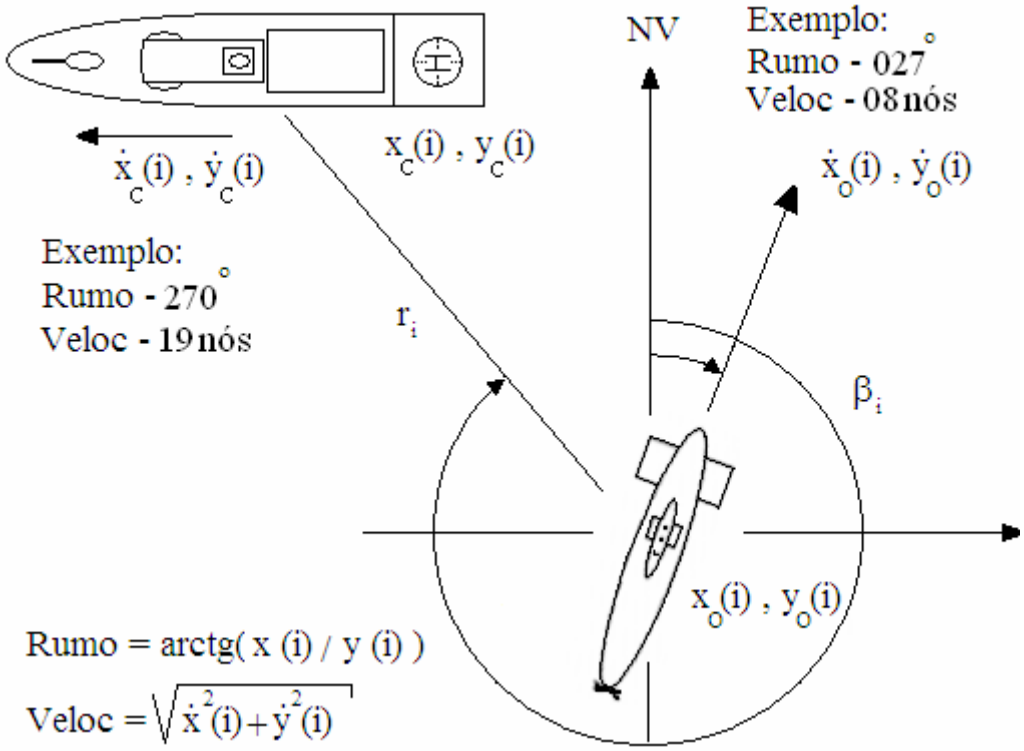


Figura 2.1: Cenário Tático de navios – Marcações, Distâncias, Rumos e Velocidades.

Os termos "estimativa de posição" ou "estimado" será empregado para definir a posição estimada do contato, em termos de Latitude e Longitude do mesmo.

Quando for empregado o termo "posição real", estaremos nos referindo à posição real do contato gerada pelo simulador de trajetórias a partir das condições iniciais e das equações do Movimento Retilíneo Uniforme (MRU) e do Movimento Retilíneo Uniformemente Variado (MRUV).

As "marcações" são os ângulos medidos a partir do NV até os sucessivos segmentos de reta imaginários que ligam o contato ao navio observador. As mesmas são obtidas indiretamente através da medida do coeficiente angular de cada um destes segmentos de reta.

O termo *Bearing-Only* (B-O) é utilizado nos problemas onde somente estão disponíveis os dados das marcações do contato tomados pelo navio observador.

2.2. Fundamentos do problema de Análise de Movimento de Alvos (TMA)

O problema de *Target Motion Analysis* (TMA) – Análise de Movimento de Alvos envolve um objeto (o navio-patrulha perseguidor) tentando calcular o curso de outro objeto (o alvo, que doravante será chamado de contato, salvo indicação em contrário) com base nos pontos ocupados anteriormente pelo contato. Quando o perseguidor se movimenta em uma linha reta, é possível determinar uma família de possíveis soluções de posições, de rumos e de velocidades para o contato. No entanto, sem o conhecimento da distância inicial entre o perseguidor e o contato, não é possível determinar qual das possíveis soluções é a correta.

Realizando uma manobra adicional o conhecimento da posição, e, portanto, da distância do contato designado, além do rumo e da velocidade do mesmo, com relação ao navio perseguidor, podem ser estimados pela maioria dos métodos matemáticos implementados por meio de algoritmos.

A figura 2.2 apresenta o movimento típico de navios durante um acompanhamento e estimação.

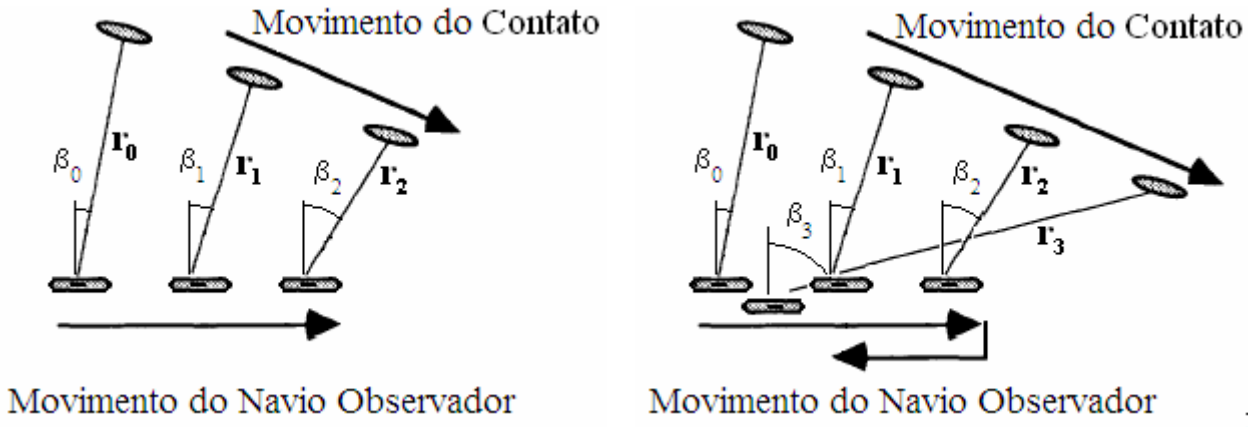


Figura 2.2: Movimento típico de navios durante um acompanhamento e estimação

Basicamente, existem dois modos de operação dos sensores existentes nos sistemas operativos das diversas plataformas navais existentes, podendo as mesmas ser aéreas, terrestres, de superfície ou submarinas:

- Modo Ativo – Neste tipo de operação, o sensor emite ondas, mecânicas no caso de sensores do tipo sonar, ou eletromagnéticas, no caso de radares. Pode-se determinar a posição exata do contato, mas a posição do observador passa a ser conhecida; e

- Modo Passivo – Para este modo de operação, o sensor somente funciona recebendo os sinais emitidos pelo contato. Somente é possível obter as marcações, que são os rumos verdadeiros, tomados relativos ao norte-verdadeiro (NV). São os rumos que o navio observador deve passar a empregar, caso deseje realizar uma interceptação do contato. As posições sucessivas do observador servem para realizar a estimativa de distância do contato. Este tipo de operação serve como forma de exercer as táticas de Dissuasão que são empregadas principalmente pelos submarinos em operação no mar. Deste modo, nega-se ao inimigo o uso do mar, pois o mesmo não sabe da existência de alguma ameaça. Um exemplo é a dos Submarinos Nucleares que navegam ao redor do globo terrestre por meses, sem que se saiba sua posição exata. Os submarinos convencionais diesel-elétricos são até mais silenciosos que estes últimos, mas possuem uma menor autonomia, precisando vir a superfície por mais vezes.

Na compreensão deste tipo de problema, são englobados assuntos relacionados com a Estimação, a Análise de Agrupamentos, a Classificação de Dados e o Apoio a Decisão, para a resolução do problema do acompanhamento de um ou mais contatos e, por conseguinte, a solução do problema do tiro de um navio observador, que pode ser tanto um navio de superfície quanto um submarino.

As técnicas empregadas na estimativa da posição dos contatos designados necessitam que sejam definidas algumas condições de contorno, de modo a possibilitar que seja obtido um resultado aproximado com um erro aceitável. Estas condições serão apresentadas nos capítulos seguintes, sendo que a seguir serão citadas apenas as técnicas normalmente empregas nos problemas de TMA.

2.3. Algoritmos Empregados na Estimação da Posição e da Velocidade do Contato

2.3.1. Método da Distância EKELUND

O Método da Distância de EKELUND [1] é utilizado para se obter uma aproximação de um intervalo de distâncias, cujos extremos são a distância mínima e a distância de EKELUND, para uma marcação final estimada do contato, após a realização de duas corridas ou pernadas. O mesmo foi criado por John Ekelund, oficial da Marinha Americana, graduado em Engenharia e que atualmente é professor associado da *Naval Postgraduate School* em Monterey na Califórnia.

O cálculo do alcance estimado do contato é possível se for considerada que a trajetória do mesmo, movendo-se supostamente em velocidade constante, possa ser aproximada por arcos de circunferências, desde que as linhas de base geradas pelo movimento do navio-patrulha perseguidor sejam muito menores do que a faixa de alcance de trabalho, ou seja, a distância entre o contato e o navio observador deve ser muito maior do que a distancia percorrida pelo navio observador durante o intervalo de medição.

Os dados resultantes que forem processados podem permitir determinar melhores cinemáticas de movimentos por parte dos navios perseguidores. Isto seria possível através de um estudo de novos conjuntos de dados de movimentos de contatos com relação ao navio observador, utilizando-se o parâmetro indicador do "grau de observabilidade do alvo", conforme a equação abaixo.

$$\Delta = |\xi OPT - \xi^{\otimes}| \tag{2.1}$$

Onde ξOPT é o melhor instante de tempo para o alcance estimado, enquanto que $\xi^{\otimes}$ é o instante de tempo em que ocorreu a mudança de leitura das marcações do alvo.

O mesmo indica o quanto o contato pode ser observado pelo navio observador. Quanto menor for este parâmetro, melhor será a observabilidade do contato em questão.

2.3.2. Algoritmos baseados em Sistemas Fuzzy

Os algoritmos derivados da Lógica Fuzzy [2,3,4] permitiram uma melhor compreensão da performance de um algoritmo de TMA do tipo *Bearing-Only* (B-O) uma vez que a interpretação dos dados, que antes dependia exclusivamente dos conhecimentos adquiridos pelos operadores dos sistemas de sonar e do seu grau de adestramento, passaram a ser auxiliadas por algoritmos baseados em Sistemas Fuzzy.

Estes permitem que o apoio a tomada de decisão seja executado através não somente do conhecimento qualitativo, mas também do conhecimento quantitativo de diversos parâmetros.

A Lógica Fuzzy ofereceu uma nova opção de tecnologia para a administração automatizada da incerteza no processo de integração de dados, uma vez que existem informações provenientes de diversos sensores existentes a bordo.

2.3.3. Algoritmos Estimadores Pseudolineares

Os Estimadores Pseudolineares (PL) [5] utilizam a minimização forçada de mínimos quadrados, que também é conhecida como método dos mínimos quadrados com restrição (CLS).

Os dados de entrada no algoritmo são as sucessivas posições do navio observador, bem como as marcações do mesmo com relação ao contato.

A formulação de um algoritmo pseudolinear determina como resultado de saída do mesmo um estimator linear. A aplicação do método dos mínimos quadrados com restrição nas equações pseudolineares com parâmetros desconhecidos consegue realizar uma linearização parcial de um problema não-linear.

A solução encontrada é algébrica e remove o ruído existente e que se encontra somado ao sinal, através da resolução de uma equação de álgebra linear sobre um par de matrizes, que são: a matriz de correlação aumentada; e a matriz restrita ou limitada.

O estimator resultante desta equação linear é empregado na resolução de um problema de autovalores e de autovetores generalizados. A solução proposta não requer conhecimento das condições iniciais, uma vez que uma parte dos dados amostrados é empregada na inicialização do algoritmo.

2.3.4. Filtros de Kalman e Filtros de Kalman Estendidos

Os Filtros de Kalman e os Filtros de Kalman Estendidos - *Extended Kalman Filters* (EKF) [25,27] são amplamente utilizados nos problemas em que se deseja estimar a posição de um contato, sabendo-se de antemão o rumo e a velocidade do navio observador, bem como a posição do contato, através do conhecimento da distância deste ao navio observador e de sua marcação com relação ao mesmo.

Normalmente, os dois últimos tipos de dados são obtidos a partir de sensores ativos como no caso de radares e sonares, quando os mesmos operam neste modo. No entanto, este não é o tipo de dados que se tem disponível no sistema em questão, uma vez que o sonar estará operando em modo passivo, e somente teremos a nossa disposição os dados de marcação.

Foi encontrado um tipo de abordagem [26] para o problema utilizando somente marcações, sem o conhecimento das distâncias através do emprego de filtros (EKF) e de uma variante chamada *Unscented Kalman Filters* (UKF), mas o mesmo requer o uso de, no mínimo, duas plataformas de sensores localizadas em posições diferentes, o que não é possível para o caso dos sistemas reais existentes e disponíveis para teste.

2.4. Transformadas *Wavelets* aplicadas à redução de ruído (*denoising*)

Em Processamento de Sinais [12] existem diversas técnicas que podem ser empregadas para melhorar a qualidade de um sinal que foi contaminado por diversos tipos de ruído. De modo geral, as mesmas realizam o *Denoising* [11,13,14] do sinal original recuperando a qualidade do mesmo até um certo ponto, mas sempre existe algum tipo de perda de informação neste tipo de processamento.

Das técnicas que foram estudadas, optou-se por empregar as Transformadas Discretas de *Wavelets* (DWT) [10] para a redução do ruído na série temporal das marcações obtidas da simulação do movimento do contato com relação ao navio observador. Esta opção se deve ao fato do processamento empregando *Wavelets* em *Denoising* ser rápido e direto e por existirem funções já previamente criadas no pacote do *MATLAB* [40].

Os algoritmos que são baseados em funções *wavelets* processam dados em diferentes escalas ou resoluções. Independente do fato da função de interesse ser: uma imagem 2D; uma curva de uma função cuja forma analítica é desconhecida; ou uma superfície topográfica. Assim, as *wavelets* oferecem uma técnica elegante na representação dos níveis de detalhes presentes em um sinal qualquer que é formado pela composição de diversos sinais ou de um sinal que possui ruído adicionado.

Elas constituem uma ferramenta matemática para decompor funções hierarquicamente, permitindo que uma função seja descrita em termos de uma forma grosseira, mais outra forma que apresenta detalhes que vão desde os mais grosseiros, aos mais finos e delicados.

Assim, podemos dizer resumidamente, que as *wavelets* são capazes de separar em diversos níveis um sinal formando uma aproximação e diversos detalhes, que são, grosso modo, os diversos níveis de ruído que foram somados ao sinal original. A figura 2.3 mostra a decomposição de um sinal em quatro níveis, cujo processo é denominado análise de *wavelet*.

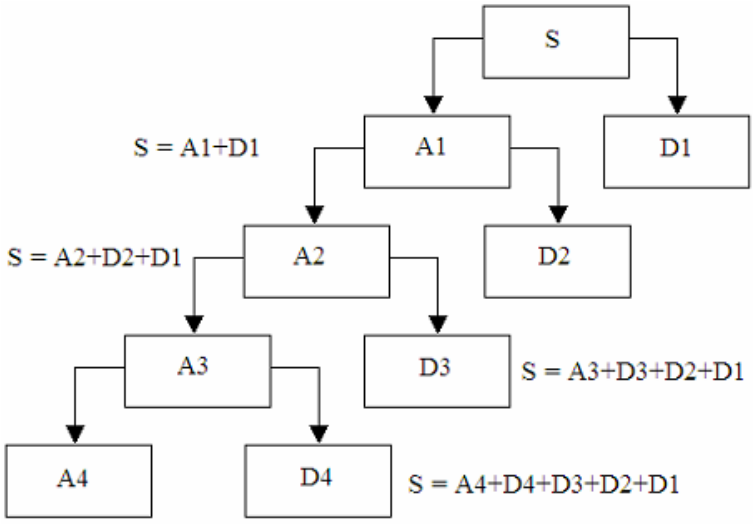


Figura 2.3: Decomposição de um sinal em quatro níveis - análise de *wavelet*

O resultado na análise de *wavelet* é **"ver a floresta e as árvores"** [38]**.** Seria como usar um equipamento ótico que tem a função de microscópio e de binóculo simultaneamente. Selecionando-se uma função, podemos ver os detalhes microscópicos e alterando a mesma, passamos a ver somente o todo, ou seja, o macroscópico.

Uma abordagem alternativa para as *wavelets* é pensar nos processos de análise e síntese de bancos filtros criticamente decimados [11], pois para que a transformada de *wavelet* exista é necessário que o banco de filtros seja de reconstrução perfeita.

Os coeficientes dos filtros têm uma relação direta com as funções de escalamento e com a *wavelet.*

Isto fez com que as *wavelets* se tornassem uma ferramenta matemática poderosa para aplicação em diversas áreas de conhecimento. Este tipo de técnica é utilizada para retirar o ruído e melhorar a qualidade de imagens, sons, sinais elétricos, séries temporais de eventos, etc.

2.4.1. Transformadas *Wavelets* – o que são?

A transformada *wavelet* [11,12,13,14,15,16] de uma função pertencente ao espaço das funções quadraticamente integráveis, $L^2\{\Re\}$, é a sua decomposição numa base formada por:

- Expansões;
- Contrações; e

- Translações.

As mesmas são geradas a partir de uma única função-mãe ψ(t), chamada de *wavelet*-mãe. Além disso, possuem determinadas propriedades que as tornam adequadas a servirem de base para decomposição de outras funções, assim como senos e cossenos servem de base para decomposições de *Fourier*.

As *wavelets* mais usadas formam um sistema ortonormal de funções com suportes compactos construídos desta forma. Esta é a razão pela qual elas podem distinguir as características locais de um sinal em diferentes escalas e, por translações, elas cobrem toda a região na qual o sinal é estudado. São adequadas, portanto para a análise de fenômenos transitórios, não estacionários ou variantes no tempo.

Mas uma pergunta ainda persiste: O que é afinal uma *wavelet*?

Uma *wavelet* é uma função com propriedades especiais. Literalmente, o termo *wavelet* significa pequena onda. Em geral, uma *wavelet* é uma função (no domínio do tempo) que satisfaz as seguintes condições:

- Tem de ter energia finita, conforme a equação abaixo no domínio da freqüência *w*:

$$c_\psi = \int \frac{|\psi(w)|^2}{|w|} dw < +\infty \qquad (2.2);$$

- É admissível, ou seja, a *wavelet* tem de oscilar no domínio do tempo para que tenha valor médio nulo, como pode ser visto em 2.3:

$$\int \psi(t) dt = 0 \qquad (2.3); \text{ e}$$

- Tem que ser regular, porque uma *wavelet* tem de ser uma função no domínio do tempo que decaí exponencialmente para que seja um operador local no domínio do tempo e da freqüência. Assim, os coeficientes da transformada de *wavelet* diminuem rapidamente com a diminuição da escala. Isto é bom para uma aplicação onde haja compressão de dados.

Podemos concluir que uma *wavelet* deve ser pequena para satisfazer a primeira condição e deve ser oscilante para satisfazer a segunda. Uma tradução informal, que é encontrada na *Web* seria a seguinte: *Wavelet* é uma pequena (*Little*) onda (*Wave*).

Outros autores [38] também consideram que o termo *Wavelet* seria originalmente uma tradução mal feita da palavra francesa ***ondeletes***, que significa também **ondas pequenas.**

2.4.2. Famílias de *Wavelets* mais comumente utilizadas:

Haar: É a primeira e a mais simples de todas. É descontínua e equivale a *Daubechies* 1 (db1). Na figura 2.4, foi gerada uma *wavelet* de *Haar* de análise e sua função de escalamento *Haar* de análise.

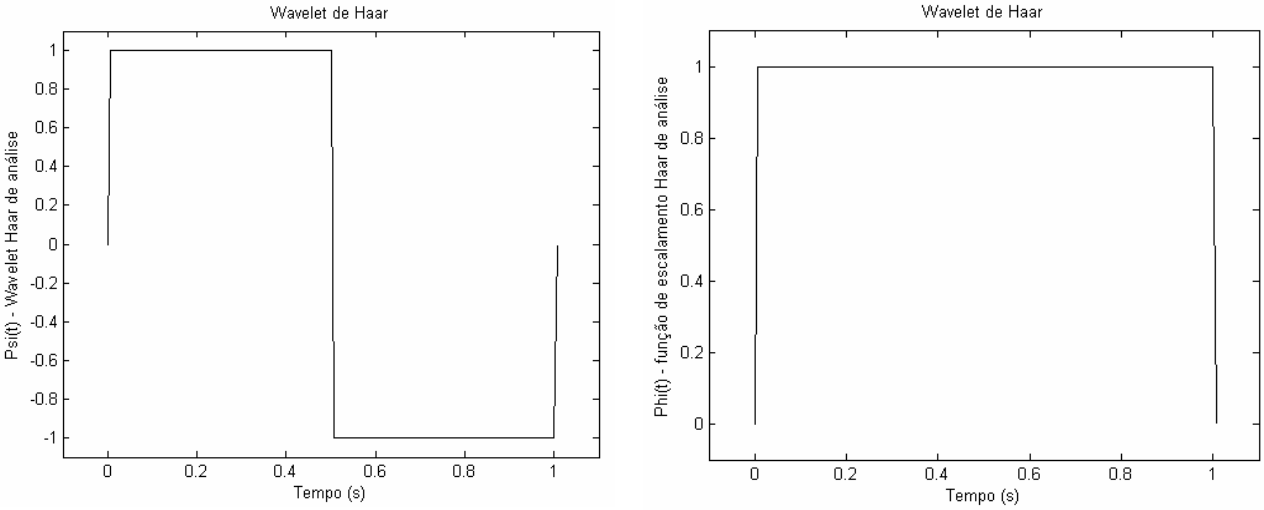


Figura 2.4: Representação de uma *wavelet* de *Haar* de análise e sua função de escalamento *Haar* de análise

A *wavelet* de *Haar* é na realidade um caso particular de um outro conjunto de *wavelets* da família conhecidas por *Daubechies* ou db. As mesmas formam um conjunto de bases ortonormais de *wavelets* suaves, com suportes compactos, ou seja, *Compactly-supported orthonormal wavelets.*

Na figura 2.5 abaixo, encontram-se representadas uma *wavelet* de *Daubechies* de nível 16 (db16) de análise e sua respectiva função de escalamento de análise.

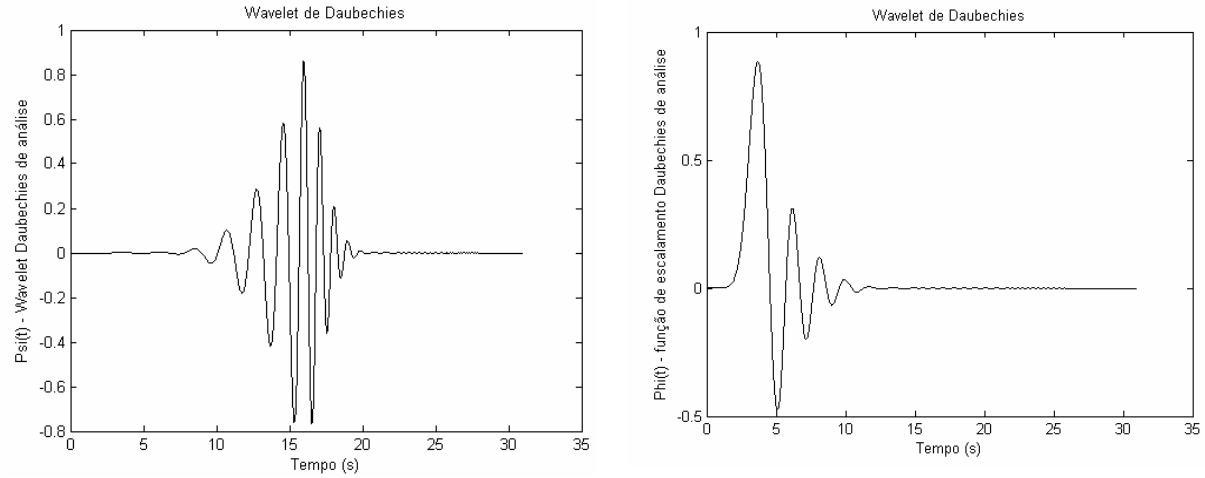


Figura 2.5: *Wavelet* de *Daubechies* de nível 16 (db16) de análise e sua respectiva função de escalamento.

Outras famílias existentes são:

Biortogonal - Apresenta a propriedade de fase linear, que é necessária na reconstrução de sinais e imagens. Utiliza duas wavelets, uma para decomposição e outra para reconstrução, o que gera propriedades interessantes;

Coiflets - A função wavelet possui 2N momentos iguais a zero e a função escala tem 2N-1 momentos iguais a zero; e

Symlets, para o *MATLAB* [39], ou Symmlets, para alguns autores [11,12,14], pois seria derivado das *symmetric wavelets* [15] - wavelets simétricas. Foi proposta como uma modificação da família Daubechies, pela própria Ingrid Daubechies, possuindo características similares as desta família. Na figura 2.6, encontram-se representadas a seguir uma wavelet da família Symmlet de nível 8 (sym8) de análise e sua função de escalamento de análise.

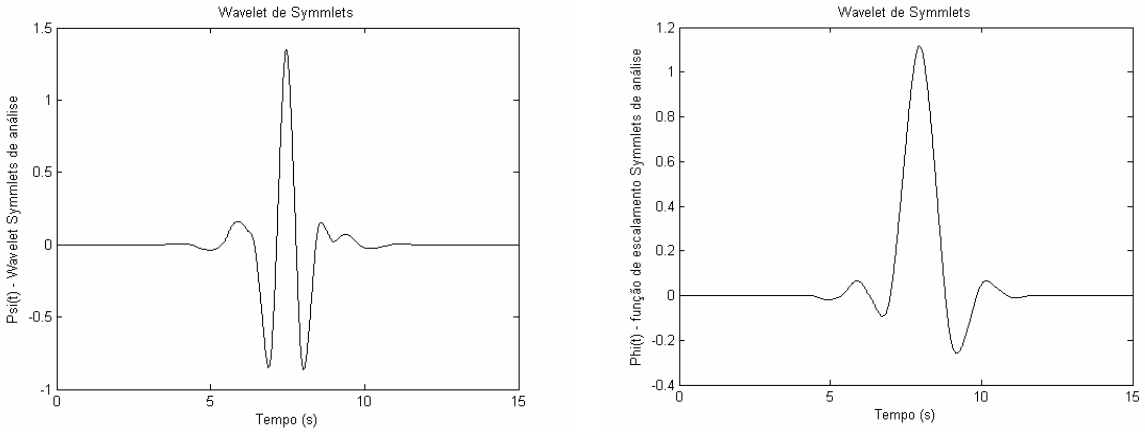


Figura 2.6: *Wavelet* da família *Symmlet* de nível 8 (sym8) de análise e sua função de escalamento

Normalmente, quando se projeta um banco de filtros utilizando as transformadas de *wavelets*, as características do sinal sobre o qual se deseja aumentar o grau de conhecimento têm uma relação direta quanto ao formato com o formato e o nível da família de *wavelets* que se deseja utilizar. Deste modo, utiliza-se uma *wavelet* que possui formato parecido com o formato do sinal desejado.

No entanto, quando se deseja realizar a filtragem de ruído aleatória, que foi adicionado ao sinal propositalmente ou não, através da utilização das *wavelets* em *denoising*, a aproximação consiste em utilizar apenas os coeficientes maiores, desprezando os menores, que seriam justamente as componentes responsáveis pelo ruído [16].

Um dos métodos mais utilizados e que utiliza este conceito é o de *Thresholding Method* que foi proposto por *Donoho* e *Johnstone* em 1994. O problema maior é o da

escolha deste nível de limiar que não pode ser nem muito grande a ponto de afetar a envoltória do sinal desejado, ou seja, o sinal que se deseja manter as características, e nem muito pequeno, a ponto de fazer com que a filtragem do ruído seja inócua.

Existem basicamente dois métodos de se calcular o *thresholding* e algumas variantes empregando a combinação heurística dos mesmos.

O método proposto por *Donoho* e *Johnstone* possui a seguinte equação para o *thresholding* λ:

$$\lambda = \sqrt{2 \cdot \sigma^2 \cdot \log n} \tag{2.4}$$

Onde σ^2 é a variância do conjunto de dados original que contém n dados.

Já o outro método é chamado minimização da função risco de Stein, que está abaixo representada:

$$S = n + a(\lambda) \cdot (\lambda^2 - 2) + \sum_{k=a(\lambda)+1}^{n} d_k^2 \tag{2.5}$$

Onde $a(\lambda)$ é o número de coeficientes menores ou iguais ao *thresholding* λ e d_k são os coeficientes das *wavelets* rearranjados em uma série crescente.

No capítulo referente a implementação do *denoising* utilizando as *wavelets*, serão vistos os métodos utilizados com o uso do *MATLAB*.

3. Revisão Bibliográfica, Comentários e Estado da Arte.

3.1. Trabalhos Pesquisados

3.1.1. Um Sistema de Esclarecimento para um Navio Patrulha com Deslocamento de 1300 Toneladas

A tese *Um Sistema de Esclarecimento para um Navio Patrulha com Deslocamento de 1300 Toneladas* é de autoria de Sílvio Fernando Bernardes Pinto, oficial da Reserva da MB que concluiu sua Dissertação de Mestrado no Instituto Militar de Engenharia em 1999.

Nesta dissertação, o autor realiza a análise de vários métodos não-recursivos cujas soluções são obtidas através do uso da regressão linear dos mínimos quadrados. Estes métodos são empregados para a dedução dos algoritmos de TMA, os quais são baseados em equações matriciais. O primeiro algoritmo analisado é o TMA de segunda ordem, que é composto de duas matrizes quadradas, cujas entradas são as medidas de marcações e de alcance do contato.

Existe também a análise do algoritmo de TMA de quarta ordem, que utiliza matrizes quadradas, e possui como entradas as marcações e os alcances do contato. A solução do algoritmo de TMA *Bearing Only* (B-O), que é analisado em seguida, é obtida também através da regressão linear dos mínimos quadrados pela manipulação de equações matriciais de ordem quatro.

É analisado um algoritmo de TMA híbrido que utiliza uma fração K das características do algoritmo de TMA de quarta ordem nas equações do algoritmo (B-O) e, segundo o autor, serve para minimizar adequadamente a soma ponderada das variâncias longitudinais e transversais do alcance.

Por fim, o autor propõe um algoritmo EKELUND automatizado, utilizando coordenadas polares modificadas, sendo que o mesmo é um caso particular do algoritmo de TMA (B-O), uma vez que para alcances longos, o valor da velocidade longitudinal (V_R) é assumido igual a zero, ficando como entradas apenas velocidades transversais ao alcance (V_{CR}) e as marcações.

Através da simulação de uma geometria típica de movimento de navio e de possíveis marcações obtidas, o algoritmo de EKELUND é sugerido como sendo o algoritmo para um Sistema de Esclarecimento de um futuro Navio Patrulha da MB. Para os dados de entrada, seriam empregadas as informações provenientes de um equipamento MAGE - Radar (Medidas de Apoio à Guerra Eletrônica) que utilizaria um algoritmo de predição de alcance.

Na dissertação acima mencionada, o autor utiliza apenas um tipo de cinemática do navio-patrulha perseguidor. Este tipo de movimento é conhecido como *lead–lag* (aproximação-afastamento), sendo que em todos os 11 conjuntos de dados simulados, o navio-patrulha perseguidor tem um rumo inicial constante da primeira pernada de 040 graus com 9 nós de velocidade, manobrando e estabilizando o rumo e a velocidade entre pernadas em 4 minutos, e tomando um rumo na segunda pernada de 280 graus com 12 nós de velocidade. A duração de cada pernada nos dados simulados é de 4 minutos.

Na figura 3.1, encontramos os Rumos e Velocidades do Navio Observador que foram utilizados nas simulações das geometrias de movimento de contato para os conjuntos de dados analisados nos resultados encontrados na dissertação mencionada.

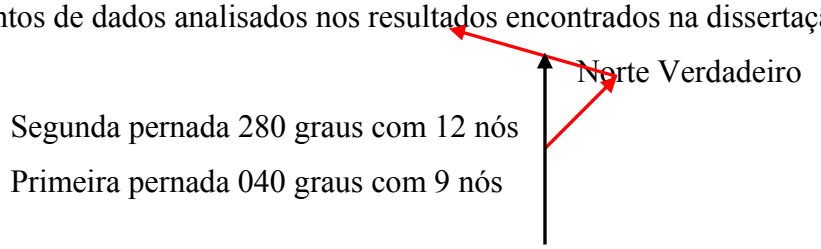


Figura 3.1: Rumos e Velocidades do Navio Observador nas simulações da dissertação em análise.

Na primeira pernada, a equação da distância mínima é obtida da equação 3.1:

$$D_{\min} = \frac{V_{crp1}}{\frac{d\beta_1}{dt}} \tag{3.1}$$

onde V_{crp1} é componente de velocidade transversal do movimento do contato com relação ao navio observador na primeira pernada e $d\beta_1/dt$ é a taxa de variação da marcação com o tempo, também para a primeira pernada. Esta taxa de variação das marcações é portanto a primeira derivada da marcação com relação ao tempo, ou seja, é a velocidade das marcações.

Já para a segunda pernada, a equação 3.2 a seguir representa a distância de EKELUND, sendo que, segundo o autor, também é conhecida como "fórmula de EKELUND":

$$D_{EK} = \frac{V_{crp1} \pm V_{crp2}}{\frac{d\beta_1}{dt} \pm \frac{d\beta_2}{dt}} \tag{3.2}$$

onde V_{crp1} e V_{crp2} são as componentes de velocidade transversal do movimento do contato com relação ao navio observador na primeira e segunda pernadas e $d\beta_1/dt$ e $d\beta_2/dt$ são as taxas de variação das marcações com o tempo, também para a primeira e segunda pernadas.

Estes valores determinam em conjunto com a marcação de referência ao final da segunda pernada o segmento de raio ($D_{EK} - D_{min}$) em que o contato está provavelmente localizado.

No trabalho desenvolvido na dissertação mencionada acima, o autor sugere que os 11 conjuntos de dados formariam "três classes" de dados, ou melhor, três conjuntos de agrupamentos de dados segundo as taxas de variação das marcações. Existiriam as seguintes classes: taxas lentas, taxas médias e taxas rápidas.

A figura 3.2, que está abaixo, representa uma possível posição do contato após o processamento com o algoritmo TMA EKELUND.

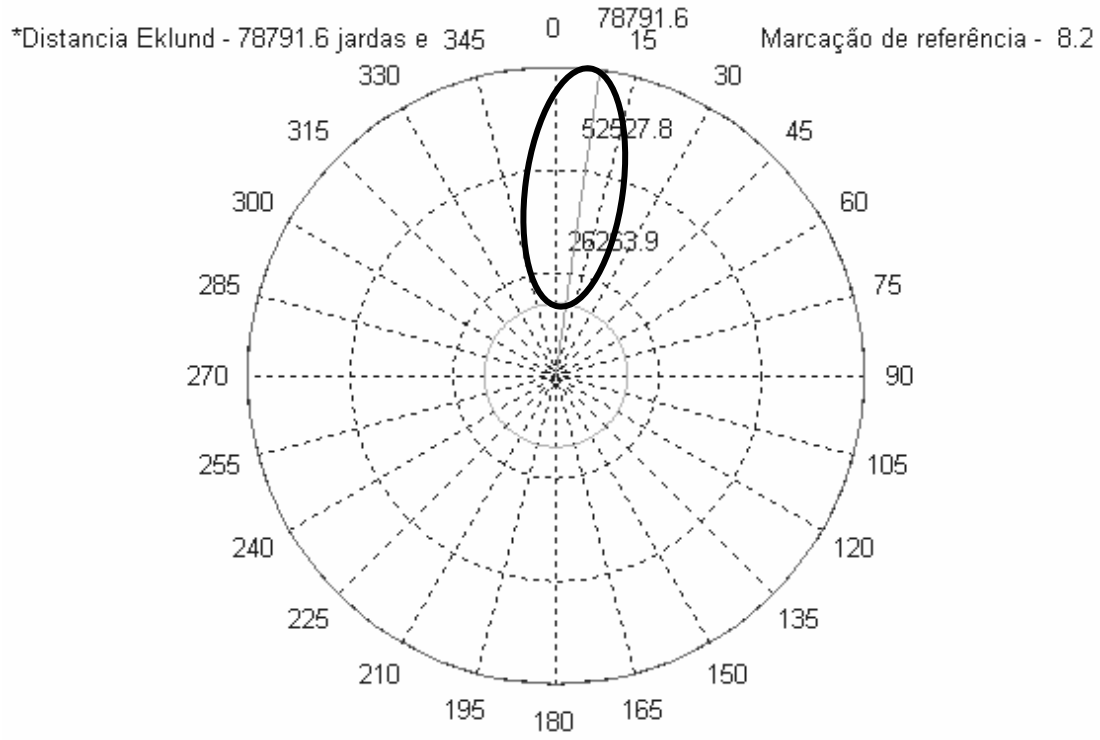


Figura 3.2: Resultados da possível posição do contato após o processamento com o algoritmo TMA EKELUND.

Os resultados obtidos pelo algoritmo TMA EKELUND para uma simulação qualquer, onde o contato está na posição inicial (X,Y) igual a (7,39) em NM com rumo 330 graus e 25 nós, e o navio observador com as condições inicias descritas conforme a figura 3.1 e partindo da origem (0,0), mostram que o contato se encontra sobre a marcação 008,2 graus a uma distância mínima de 18720 jardas e a uma distância máxima EKELUND de 78791,6 jardas. Estas 60071,6 jardas de incerteza representam cerca 29,66 NM de incerteza, ou 54,93 km. O navio patrulha perseguidor pode realizar outras corridas de busca e triangular as marcações sucessivas para melhorar a estimativa da medida.

O autor comenta que os resultados alcançados pelo algoritmo EKELUND proposto na dissertação são melhores quando estas taxas são altas, e as manobras de mudança de rumo da primeira para a segunda pernada são mais acentuadas, fazendo com que o algoritmo obtenha respostas mais precisas. Existe ainda o parâmetro que foi apresentado na equação 2.1 que indica o grau de observabilidade do contato; quanto menor for o mesmo, melhor será a observabilidade do contato em questão.

3.1.2. Fuzzy Logic-Based Information Processing in Submarine Combat Systems

O segundo artigo selecionado, *Fuzzy Logic-Based Information Processing in Submarine Combat Systems*, é de autoria de Chidambar Ganesh. O pesquisador trabalha no *Naval Undersea Warfare Center*, centro de pesquisas mantido pela Marinha Americana em *Newport*.

Neste artigo o autor comenta como os métodos atuais para integração de dados e apoio à decisão em sistemas de combate de submarino não consideram adequadamente a incerteza em modo automatizado, e por isso continuam a depositar uma confiança muito grande na interpretação humana do operador sonar que manipula os dados conforme suas experiências passadas e o treinamento recebido. Em problemas onde somente é executada uma pernada simples (*single leg*), o método não recursivo *Maximum Likelihood Estimator* (MLE) – Estimativa de Máxima Probabilidade determina uma família de curvas de possíveis soluções sobre a marcação e a distância do contato, sendo a velocidade do mesmo é considerada constante, o que torna quase impossível à determinação precisa do ponto provável de localização do mesmo.

Na figura 3.3 a seguir, o autor mostra uma família de possíveis soluções para o problema de TMA.

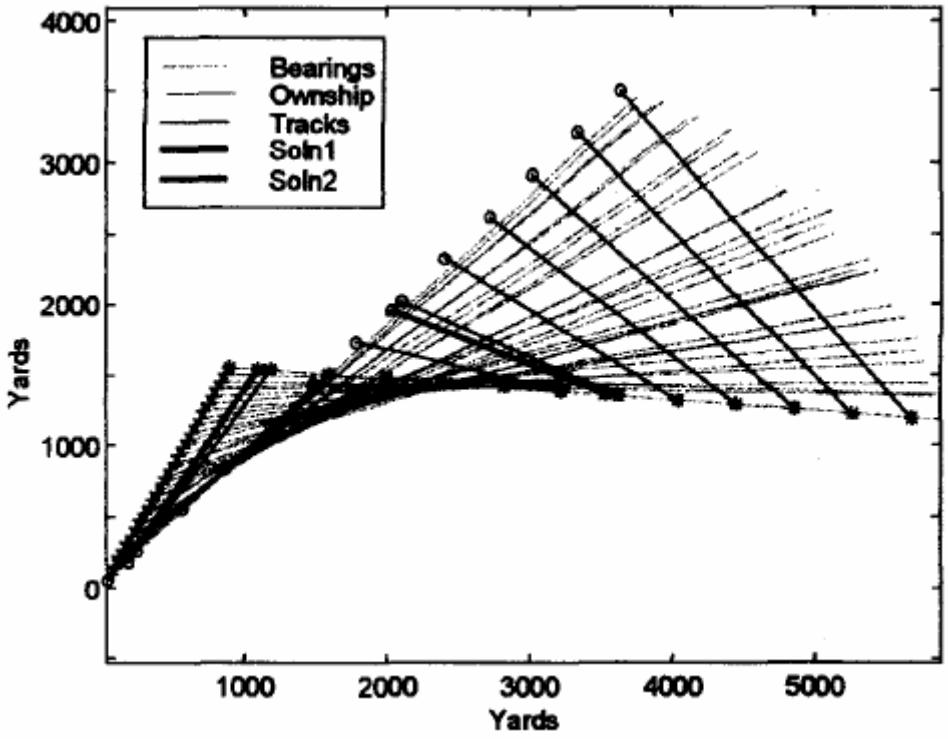


Figura 3.3: Possíveis resultados para uma problema de TMA para o paper em discussão [2].

Os sistemas de informação de combate de submarinos requerem a integração de dados, de diversas fontes para geração de cenários táticos de apoio, avaliação e planejamento, permitindo a alocação e o controle dos recursos do navio. A utilização do sistema implementado permite um ganho na medida que automatiza diversas funções até então executadas manualmente pelos operadores dos sistemas de combate, que recebem informações dos diversos sensores, dos submarinos além de melhorar a qualidade e a confiabilidade dos dados compilados para a tomada de decisão pelo comandante do submarino.

A Lógica Fuzzy oferece uma tecnologia nova para a administração automatizada da incerteza no processo de integração de dados. A aplicação desta tecnologia através de um FIS – *Fuzzy Inference System* (ou Sistema Fuzzy de Inferência), para a caracterização Fuzzy da velocidade do contato, que possui incertezas de informação, demonstra uma melhora significativa na qualidade da solução da localização do contato para o problema de TMA com pernada simples.

Na figura 3.4, está representado o FIS de velocidade implementado pelo autor no trabalho [2] em análise.

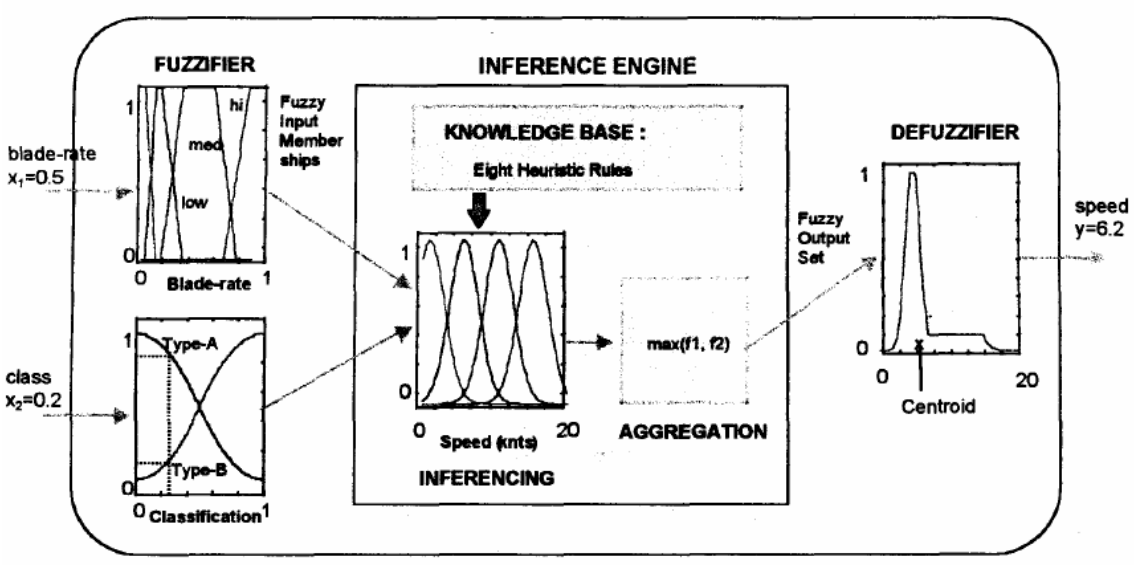


Figura 3.4: FIS – *Fuzzy Inference System* (ou Sistema Fuzzy de Inferência) implementado no artigo [2].

A incerteza na determinação da provável posição do contato é descrita por uma área na região de incerteza, e é obtida por combinação da lógica fuzzy com métodos probabilísticos convencionais (Métodos Bayesianos).

Na figura 3.5, são apresentados os resultados alcançados no trabalho [2]. Pode-se observar as regiões de incerteza que foram obtidas.

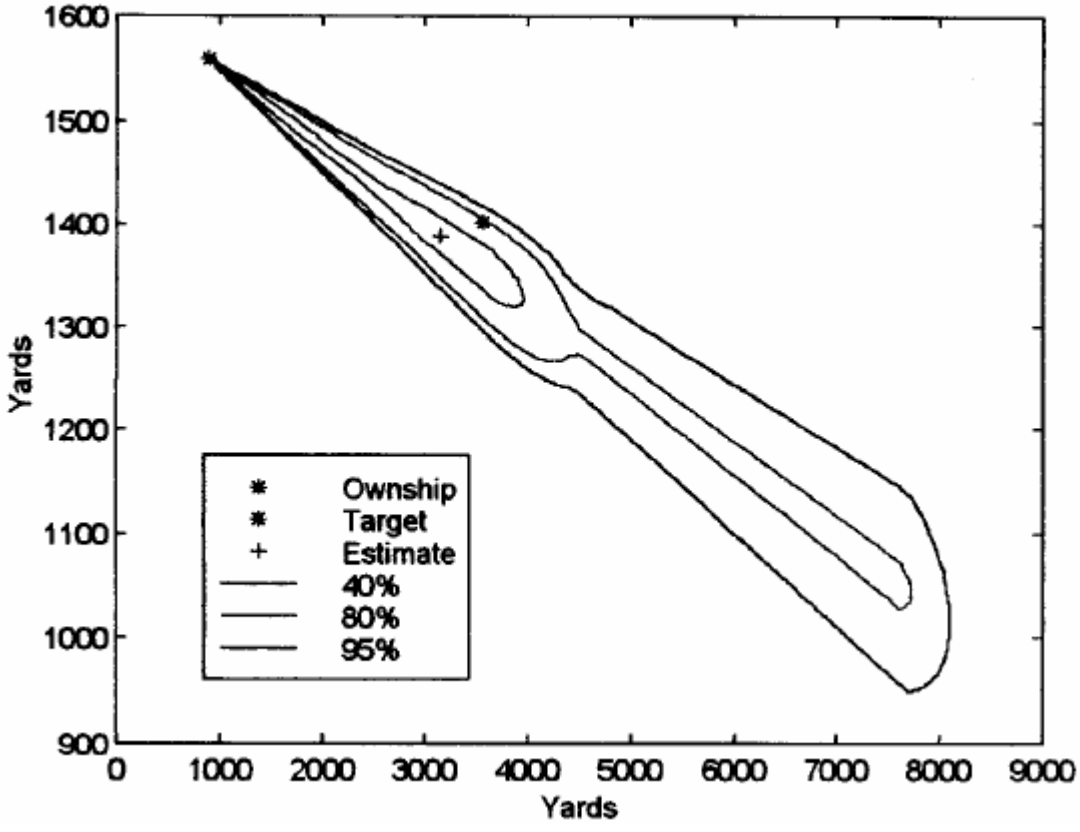


Figura 3.5: Posições Verdadeiras, Estimativas e Regiões de Incerteza do Contato e do Navio Observador [2].

Quando foi publicado este artigo, estava em curso um esforço de busca para a modelagem da incerteza dentro de uma estrutura básica de um sistema de Lógica Fuzzy, além de que estava sendo realizado um estudo para investigar a relação teórica entre a Teoria dos Sistemas Fuzzy e as teorias probabilísticas baseadas nos Teoremas Bayesianos.

Alguns fatos descritos neste artigo não ficaram totalmente entendidos, fato que fez com que um artigo anterior publicado pelo mesmo autor com outros colaboradores fosse pesquisado.

Este artigo anteriormente realizado e publicado em 1998 está descrito a seguir.

3.1.3. A Fuzzy Logic-Based Intelligent Controller for Contact Management Data Integration

O artigo, *A Fuzzy Logic-Based Intelligent Controller for Contact Management Data Integration,* foi publicado [3] em 1998 no *Fuzzy Systems Proceedings* do *IEEE World Congress on Computational Intelligence*, realizada de 4 a 9 de maio de 1998 pelo IEEE.

O mesmo é de autoria de Chung T. Nguyen, Chidambar Ganesh e Kai F. Gong, pesquisadores que trabalham no *Naval Undersea Warfare Center*, centro de pesquisas mantido pela Marinha Americana em *Newport*, *Rhode Island*, *USA*.

Neste artigo o autor descreve a implementação de um *Intelligent Process Controller* (IPC) – Controlador Inteligente de Processos, que baseado em Sistemas Fuzzy de Inferência (FIS), permite determinar uma posição estimada de um contato, para um problema de pernada simples (*single leg*), com um grau de incerteza menor do que o obtido usando as técnicas tradicionais empregadas em um problema de TMA.

Na figura 3.6, está representado o IPC proposto no artigo em análise.

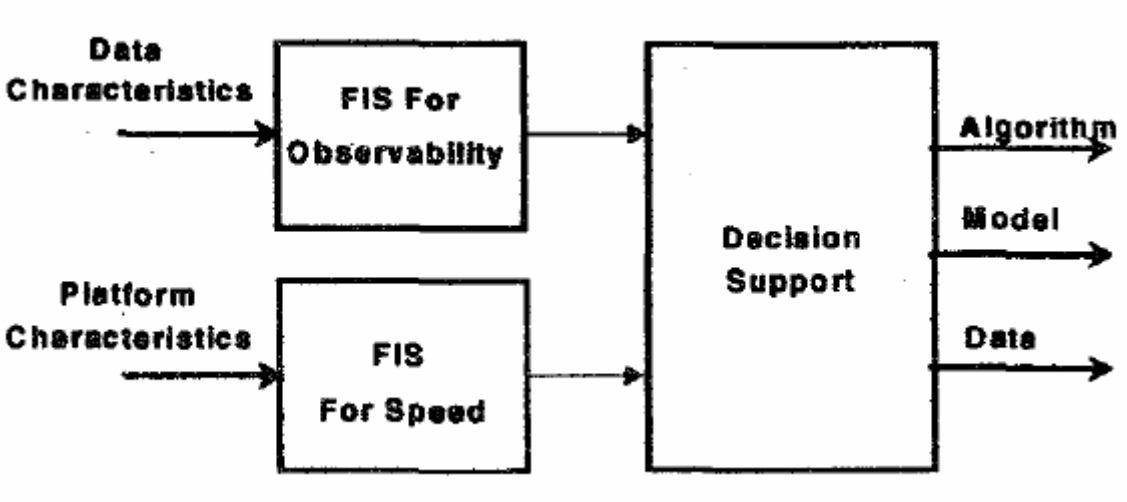


Figura 3.6: ***Intelligent Process Controller*** **(IPC) – Controlador Inteligente de Processos - proposto em [3].**

As incertezas intrínsecas, como o ruído no meio, as rápidas variações de propagação do sinal no meio líquido devido às diversas condições de temperatura da água do mar, em camadas, com a profundidade e salinidade, além das condições de fronteira entre estas camadas adiabáticas, dos sistemas de aquisição dos sistemas sensores (sonar) dos meios que operam submersos (submarinos), fazem com que os sistemas de informação de combate de submarinos tenham suas performances prejudicadas, ainda mais quando em cenários táticos onde existam múltiplos contatos, o processamento e a integração dos dados sejam mais complexos, sobrecarregando os sistemas de combate do submarino.

Basicamente são propostos neste artigo dois tipos de Sistemas Fuzzy de Inferência - *Fuzzy Inference System* (FIS), sendo um para medir o grau de observabilidade do contato e o outro para avaliação da velocidade do contato. Uma estrutura básica de um FIS está representada na figura 3.7 abaixo:

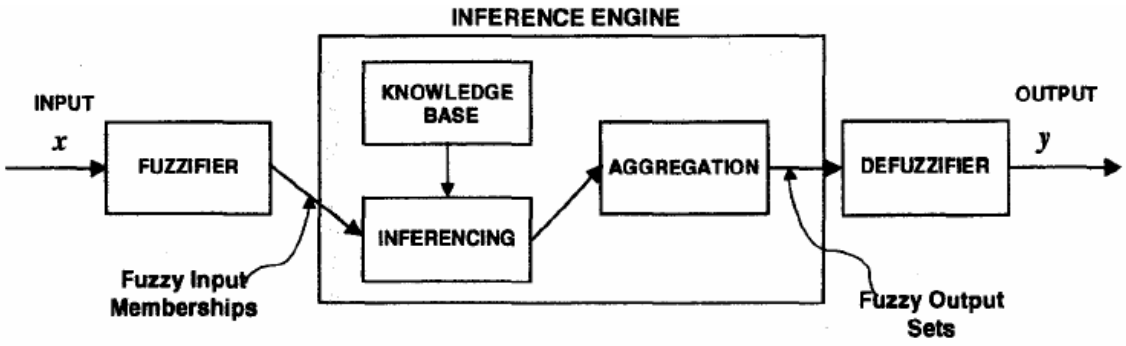


Figura 3.7: ***Fuzzy Inference System*** **(FIS) – Sistema Fuzzy de Inferência –- proposto em [3]**

O FIS de observação é caracterizado pela sua capacidade de determinar um número específico de parâmetros do contato, que é função da qualidade dos dados (*data stream*) e da qualidade do processo de amostragem dos dados, ou seja, do desvio padrão do ruído (*noise standard deviation*). O FIS de observação é construído segundo o modelo Fuzzy de Sugeno, pois este modelo é apropriado para observar classes que são descritas em termos de números inteiros de parâmetros que devem ser estimados.

A fuzzificação de algumas entradas e saídas são mostradas segundo a figura 3.8 a seguir:

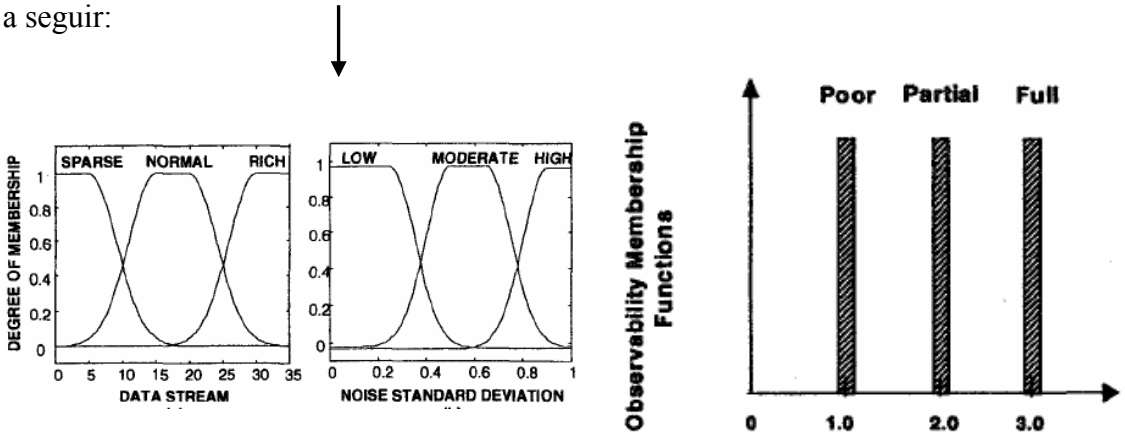


Figura 3.8: Fuzzificação de algumas entradas e saídas utilizadas no problema do trabalho [3] em análise.

O FIS de velocidade é implementado segundo o modelo Fuzzy de Mamdani, sendo as entradas a classificação do contato e o espectro normalizado, e a saída é a velocidade avaliada do contato, podendo o mesmo ser do tipo I ou do tipo II. O FIS de velocidade é empregado para determinar, a partir das informações de velocidade do contato, o modelo de cinemática de movimento do mesmo.

Na figura 3.9, estão representadas outras fuzzificações de entradas e de saídas.

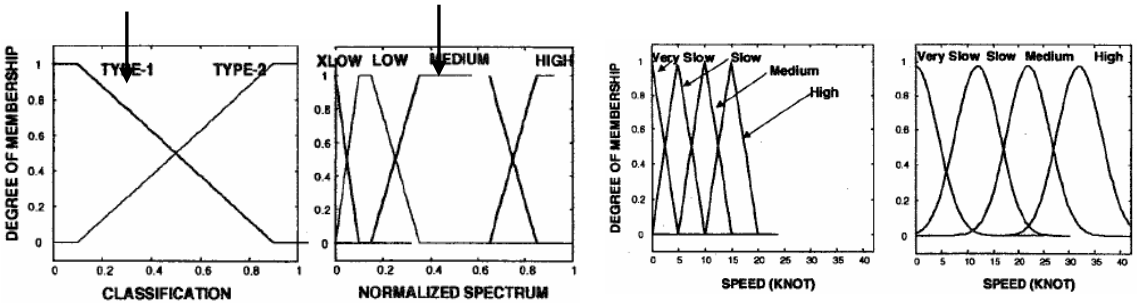


Figura 3.9: Outras fuzzificação de entradas e de saídas utilizadas no problema do trabalho [3] em análise.

Os resultados experimentais do primeiro exemplo, utilizando como conjunto de entrada dados obtidos a partir de simulação, para um problema de TMA do tipo *Bearing-Only* (B-O)– Somente Marcações, são obtidos a partir de 30 amostras de

marcações que possuem ruído com distribuição normal. Segundo está relatado, o FIS de observação analisa estes dados, que não são fornecidos, talvez por questões de confidencialidade, e determina um alto índice de observabilidade, de 2,76, indicando que os três parâmetros do problema de MLE, marcação, velocidade de marcação (ou taxa de marcação) e aceleração de marcação podem ser obtidos. Novamente, sem citar como são obtidas as soluções, um conjunto de curvas com possíveis soluções é mostrado na figura 3.10 abaixo:

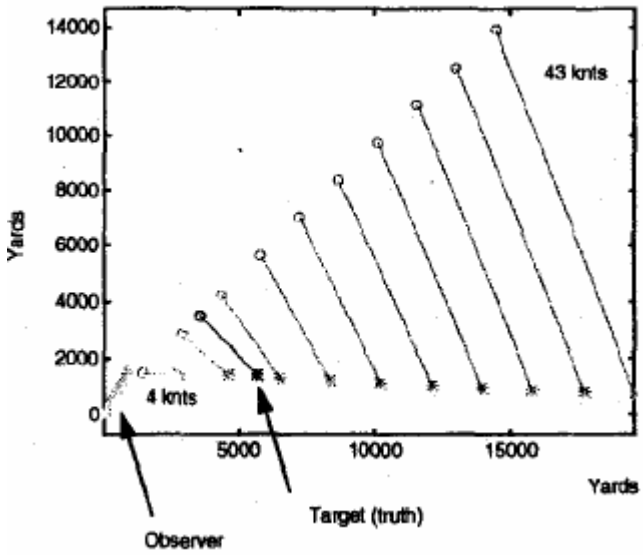


Figura 3.10: Familias de possíveis soluções de posições do contato e localização do contato e do navio observador [3].

O FIS de velocidade é utilizado para determinar a cinemática de movimento do contato, mas como novamente os dados de entrada não são disponibilizados, os autores somente informam que para o cenário descrito a entrada de classificação é igual a 0,2, o que indica que o movimento do contato é lento e mais próximo de uma plataforma do tipo I. O espectro normalizado é igual a 0,5, valor que também não se sabe de onde e como foi obtido, e segundo a função de pertinência correspondente, seria indicado que o contato está se deslocando segundo uma velocidade mediana.

Por fim, os autores afirmam que a velocidade do contato pode ser caracterizada por uma função de pertinência Fuzzy conforme descrito na figura 3.11 seguinte, mas sem mostrar como foi obtido este resultado:

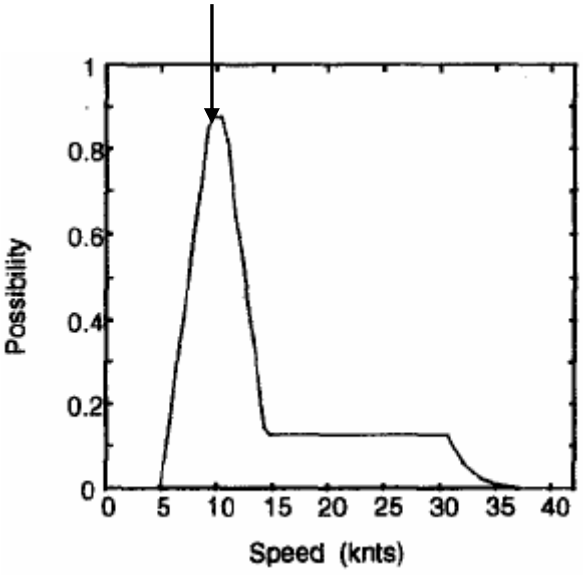


Figura 3.11: Velocidade do contato pode ser caracterizada por uma função de pertinência Fuzzy – artigo [3].

Segundo os autores, esta curva característica pode ser utilizada como um peso fuzzy associado com diferentes trajetórias do contato e pode ser integrada a solução do problema de TMA do tipo pernada simples (*single leg*), da figura 3.10.

A figura 3.12 mostra a curva que segundo os autores corresponde a uma melhora na incerteza dos resultados, pois os pesos determinam que o contato estaria localizado em um intervalo de 3 a 6 kjardas, para uma velocidade estimada de 10 nós.

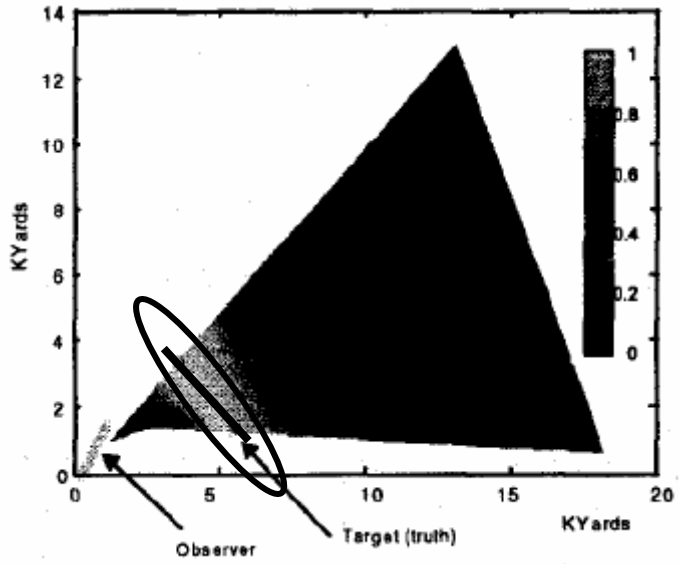


Figura 3.12: Resultado após o processamento Fuzzy utilizando os pesos determinados – artigo [3].

O segundo exemplo experimental também não mostra os dados de entrada no FIS de velocidade, apenas informando que o resultado da fuzzificação estão caracterizados pela figura 3.13 a seguir:

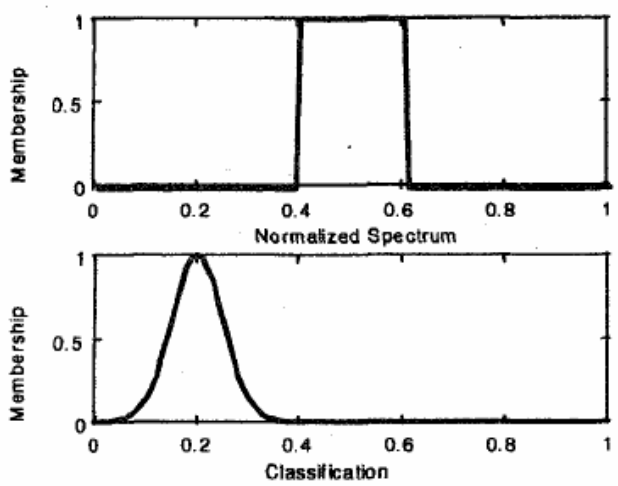


Figura 3.13: Resultado da caracterização dafuzzificação das incertezas de entrada pelo FIS – artigo [3].

O resultado da saída de velocidade após a defuzzyficação está na figura 3.14:

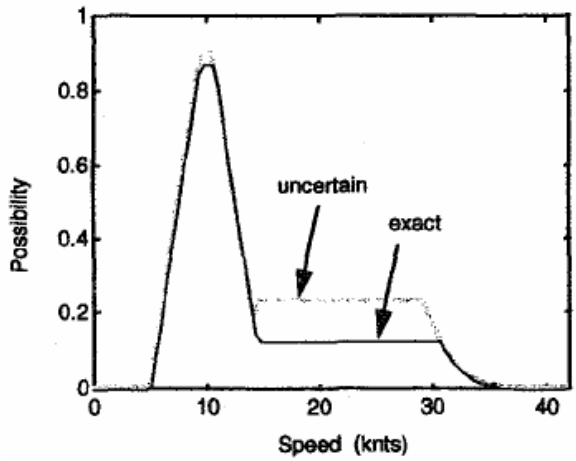


Figura 3.14: Resultado da saída de velocidade após a defuzzyficação – artigo [3].

O grande problema para utilizar as idéias descritas neste artigo, para os dados experimentais existentes nesta dissertação diz respeito ao fato de que não existem algumas informações que foram, muito provavelmente de maneira proposital, omitidas, entre as quais podemos citar as bases de regras.

Por isso, optou-se em tentar melhor compreender as informações e os dados existentes e o funcionamento de outros algoritmos de TMA do tipo *Bearing-Only* (B-O), de forma a permitir criar no futuro um novo algoritmo empregando técnicas de Inteligência Artificial e talvez outros métodos, como Estimadores Pseudolineares (PL), os Sistemas Fuzzy e os Filtros de *Kalman*. A tentativa de fazer um sistema próximo do que foi implementado nestes dois primeiros artigos demandaria um tempo muito grande e que poderia não lograr êxito, devido às informações existentes e aos dados disponíveis nos sistemas e equipamentos em uso.

3.1.4. Fuzzifying a Target Motion Analysis Model Using Fril and Mathematica

Este artigo, denominado *Fuzzifying a Target Motion Analysis Model Using Fril and Mathematica,* foi publicado [4] em 1994 no Terceiro *Fuzzy Systems Proceedings* do *IEEE World Congress on Computational Intelligence*, realizada de 26 a 29 de junho de 1994.

O mesmo é de autoria de J.F. Baldwin e T. P. Martin, pesquisadores que trabalham no *Advanced Computing Research Centre* do *Department of Engineering Mathematics* da *University Walk, Bristol, UK*.

Neste artigo, os autores descrevem como implementaram usando *Fril* e *Mathematica* um algoritmo TMA baseado em lógica fuzzy.

Fril é uma linguagem de programação lógica que incorpora métodos sofisticados por controle fuzzy e incerteza probabilística. Uma interface baseada no conhecimento, o Manual Inteligente, foi escrita em *Fril* para uso como ferramenta técnica de documentação e software para modelagem matemática.

Neste artigo, os autores mostram como o Manual Inteligente pode ser usado para investigar dois modelos matemáticos fuzzy em um problema de análise de movimento do alvo (TMA), e implementados com o uso do programa *Mathematica*. Deve ser observada especial atenção quando são executados cálculos que envolvem quantidades fuzzy, e o Manual Inteligente, implementado em *Fril*, é capaz de analisar os modelos e determinar quais métodos de cálculo fuzzy são mais apropriados.

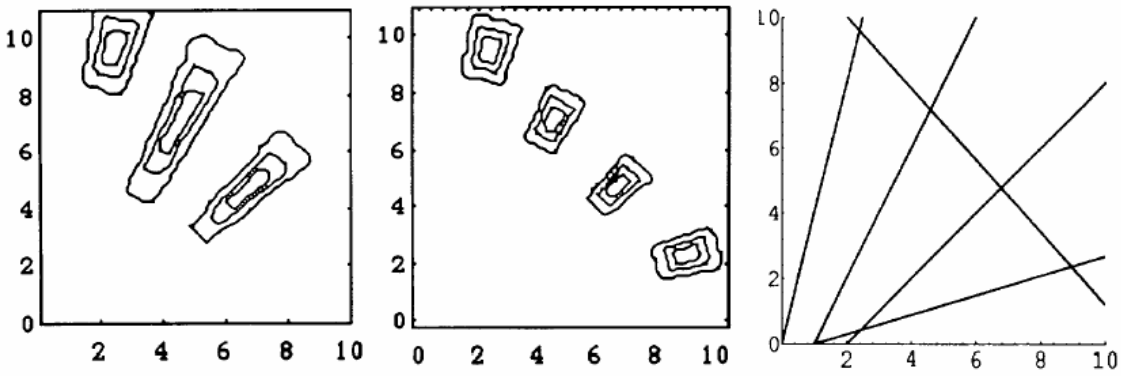


Figura 3.14: Resultados de possíveis posições do contato após o processamento com o algoritmo Manual Inteligente que foi implementado no trabalho em análise em *FRIL* –artigo [4].

3.1.5. *An Asymptotically Unbiased Estimator for Bearings-Only and Doppler-Bearing Target Motion Analysis*

O artigo seguinte, *An Asymptotically Unbiased Estimator for Bearings-Only and Doppler-Bearing Target Motion Analysis,* de autoria de K. C. Ho e Y. T. Chan, *Senior Members* do IEEE, tem por objetivo propor uma nova técnica baseada em estimadores Pseudolineares [5].

A técnica de *Bearings-only* (B-O) empregada nesse artigo para a análise de movimento do alvo (TMA) tenta obter uma trajetória estimada do contato tendo somente como entradas as marcações do contato com relação ao navio observador e as sucessivas posições deste navio ao longo do intervalo de tempo de medição. Nesta técnica, o equipamento de sonar, por exemplo, estará operando em modo passivo, ou seja, estará operando apenas em recepção, sem emitir pulsos.

Já no caso da técnica de *Doppler-bearing* (D-B), além das entradas que foram utilizadas no caso anterior, utilizam-se também as medidas da freqüência do sinal de tom característico do contato, e que sofre variação em freqüência devido ao efeito *Doppler*, além das velocidades do navio observador em cada instante de tempo de amostragem.

Os autores passam então a denominar as técnicas B-O e D-B, empregadas em problemas de TMA, por BO-TMA e DB-TMA, respectivamente.

Os problemas de BO-TMA e de DB-TMA são não-triviais porque as equações de medida são não-lineares com relação aos parâmetros do contato designado.

O trabalho propõe um estimator assintoticamente imparcial para cada um dos problemas de TMA em questão, sendo que deste modo, seria conseguida uma maior imunidade ao ruído inerente aos processos de medida.

Para atingir o objetivo proposto acima, são empregados os Estimadores Pseudolineares (PL) que utilizam a minimização forçada de mínimos quadrados, que também é conhecida como método dos mínimos quadrados com restrição (CLS).

A formulação de um algoritmo pseudolinear determina como resultado de saída do mesmo um estimator linear. No entanto, a estimativa resultante é parcial. Os autores comentam que o comportamento da convergência não é facilmente controlável.

O método proposto aplica esta minimização forçada de mínimos quadrados nas equações pseudolineares com parâmetros desconhecidos. Com isso, consegue-se uma linearização parcial de um problema não-linear.

A solução proposta é algébrica e remove o ruído existente e que se encontra somado ao sinal, através da resolução de uma equação de álgebra linear sobre um par de matrizes, que são: a matriz de correlação aumentada; e a matriz restrita ou limitada.

O estimator resultante desta equação linear é empregado na resolução de um problema de autovalores e de autovetores generalizados.

A solução proposta não requer conhecimento das condições iniciais do contato e do navio observador, uma vez que uma parte dos dados amostrados é empregada na inicialização do algoritmo. O mesmo também não apresenta, segundo os autores, problemas de convergência.

São apresentados de maneira abreviada os algoritmos propostos para BO-TMA e DB-TMA.

Os algoritmos apresentados melhoram, segundo os autores, a precisão de estimação quando uma nova medida é adquirida, não requerendo uma nova decomposição de autovalores generalizada para atualização da solução.

São apresentados alguns resultados de simulações utilizando-se de um movimento de *zig-zag* com mudanças de rumo de 90 graus para o navio observador, nos instantes de tempo que obedecem a seguinte equação:

$$t = 50 + 100 \cdot i \ , \text{ onde } i = 0,1,2,3.... \tag{3.3}$$

O navio observador parte da posição inicial (X , Y) correspondente a origem (0 , 0), com rumo inicial 000, que corresponde ao NV, e velocidade constante de 12,7 m/s, que corresponde a 45,72 km/h, ou ainda a 25,4 nós.

Já o contato parte da posição inicial correspondente a (12,7 km, 12,7 km), o que corresponde também à posição (7 NM, 7 NM), segundo um rumo verdadeiro de 030 e velocidade de 9 m/s, equivalente a 32,4 km/h, ou 18 nós.

O período de amostragem dos dados T é igual a 2 segundos, para um tempo total de simulação de 1000 segundos, correspondendo portanto a 500 iterações.

Para inicialização dos algoritmos são usadas as 50 primeiras amostras, o que corresponde a 10% dos dados totais. Além disso, é utilizada uma constate η igual a 10^{-10} como critério de parada, quando se chega a uma convergência satisfatória, no *loop* mais interno do algoritmo.

O estimator proposto alcança resultados melhores ou iguais ao CRLB, ao estimador ML - *Maximum Likelihood Estimator* e a técnica IV - (*Instrumental Variable* – IV). para um conjunto de dados com ruído branco Gaussiano adicionado.

3.1.6. Recursive Bayesian Inference On Stochastic Differential Equations

Essa tese de Doutorado [26], intitulada *Recursive Bayesian Inference On Stochastic Differential Equations,* foi defendida em 2006 por Simo Särkkä na *Helsinki University of Technology*, localizada na Finlândia. Corresponde a um dos mais recentes trabalhos existentes sobre estimação Bayesiana recursiva utilizada em problemas que envolvem sistemas dinâmicos não-lineares, os quais podem ser modelados como equações diferencias discretas com observadores estocásticos.

Os algoritmos recursivos de estimação em tempo-real para estes problemas de filtragem contínua-discreta são chamados tradicionalmente de *optimal filters* (filtros ótimos) e os algoritmos recursivos que calculam as estimativas baseadas em grupos de observações são chamados *optimal smoothers* (polidores ou suavizadores ótimos).

Na tese em análise são apresentados novos algoritmos práticos que realizam assintoticamente a aproximação de *optimal filters* e *optimal smoothers.*

As formulações matemáticas apresentadas ao longo da tese são probabilísticas e são apresentados algoritmos de estimação baseados nas teorias Bayesianas. Isto significa que os parâmetros não conhecidos, as funções desconhecidas e os processos físicos com ruído são tratados como processos aleatórios no mesmo conjunto-espaço de probabilidade. As aproximações Bayesianas produzem um modo consistente de computar a filtragem ótima e as estimativas suaves, pois as suposições do modelo determinam um bom modo de analisar e tratar as incertezas.

As equações formais dos filtros ótimos Bayesianos, no domínio contínuo-discreto, e soluções suavizadas são bem conhecidas, mas as soluções analíticas exatas são disponíveis somente para os modelos de Gaussianos lineares e para alguns outros casos especiais restritos. As principais contribuições do trabalho de doutorado de Simo Särkkä são mostrar como os recentes desenvolvimentos no domínio discreto do tempo dos *Extended Kalman Filters* (EKF), dos *Unscented Kalman Filters* (UKF), dos filtros de partícula, e os correspondentes *smoothers* podem ser aplicados no cenário contínuo-discreto.

Os desempenhos de estimação dos filtros EKF, dos novos filtros UKF e dos *smoothers* são comparado através de simulações. Soluções baseadas na filtragem contínuo-discreta também são apresentadas para os problemas de localização de um número desconhecido de contatos, do cálculo da expansão de uma doença infecciosa em uma população e da predição de uma série temporal desconhecida.

No caso de múltiplos contatos, são usados mais de dois sistemas de sensores fixos localizados em diferentes posições e coletando e transmitindo os dados de marcações do contato em tempo real, conforme ilustrado na figura 3.15, que está no trabalho analisado.

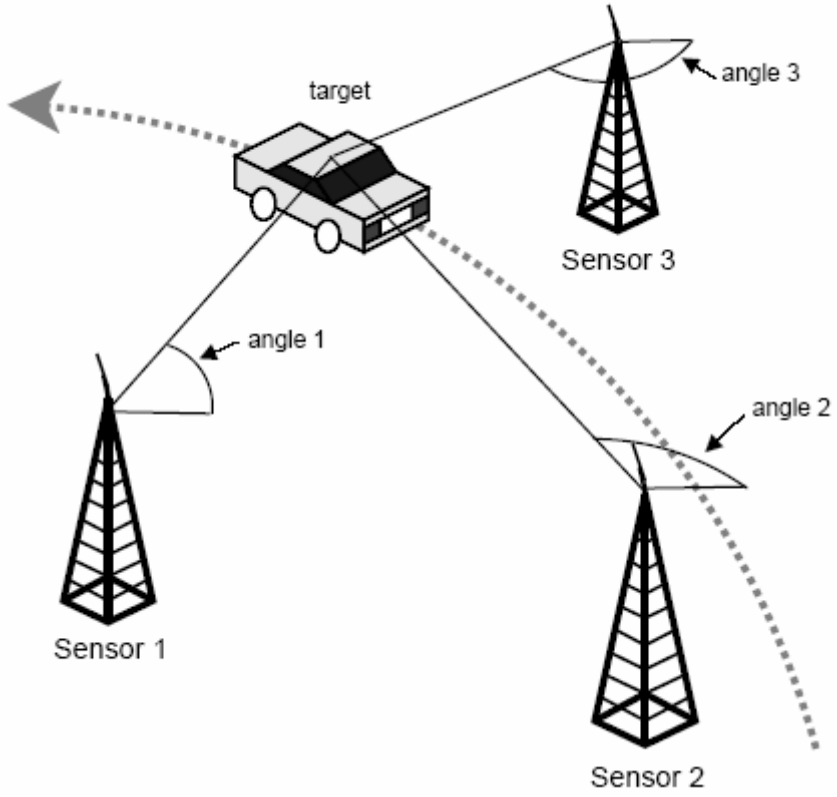


Figura 3.15: Exemplo de estimação da posição do contato usando filtros EKF, UKF e *Smoothers* – Tese [5].

3.2. Comentários adicionais sobre a revisão bibliográfica e o estado da arte

Ao final da revisão bibliográfica, pode-se chegar a algumas conclusões:

- O algoritmo EKELUND realiza a estimação da posição somente ao final da segunda pernada e possui uma incerteza relativamente grande, pois a mesma está compreendida entre a distância mínima, obtida após a

primeira pernada, dada pela equação 3.1 e pela distância de EKELUND, conforme visto na equação 3.2;

- Os algoritmos baseados em Lógica Fuzzy são de difícil implementação, pois dependem de características relacionadas aos processos de medida, e que não estão disponíveis no momento;
- A técnica de *Bearing-Only* (B-O) que emprega os Estimadores Pseudolineares (PL) é dependente somente das marcações e das posições do navio observador. Por isso, pode ser implementada sem a necessidade de maiores conhecimentos dos sensores do navio observador. A técnica de *Doppler-bearing* (D-B) utiliza além dos dados de marcação e posição do navio observador, um outro tipo de dado obtido do equipamento sonar, que é a medida da freqüência do sinal de tom característico do contato, e que sofre variação em freqüência devido ao efeito *Doppler*. Por isso, possui a mesma dificuldade de implementação dos algoritmos que empregam Lógica Fuzzy; e
- Nas técnicas estudadas que utilizam *Filtros de Kalman* e suas variantes, são usados mais de dois sistemas de sensores fixos localizados em diferentes posições e coletando e transmitindo os dados de marcações do contato em tempo real. Este tipo de implementação também apresenta restrições de ordem técnica.

Os resultados obtidos pelo algoritmo EKELUND, que foi tratado pela tese revisada [1], e pela técnica de *Bearing-Only* (B-O), empregando os Estimadores Pseudolineares (PL) no trabalho original [5], permitem concluir que a implementação dos mesmos seria tecnicamente viável. Esta afirmação é verdadeira pois os tipos de dados de entrada que estão disponíveis, marcações do contato e velocidades do navio observador, para a primeira, e marcações do contato e coordenadas de posição do navio observador, para o segundo, são compatíveis com os dados reais existentes, sendo necessário repetir os resultados alcançados para a validação das técnicas estudas. Posteriormente, poderão ser propostas outras técnicas que melhorem os resultados alcançados até então.

Embora já tenham sido citadas ao longo do texto algumas referências bibliográficas, bem como apresentados alguns conceitos importantes, persiste a necessidade de que sejam feitos comentários e observações adicionais, ainda que de

forma sucinta, sobre a revisão bibliográfica do tema da tese, bem como sobre o estado da arte.

Considerando a realidade brasileira sobre o assunto, as principais contribuições sobre o tema são encontradas em trabalhos e teses de algumas universidades e centros de pesquisa que são ligados as Forças Armadas Brasileiras (FFAA), como o IME e o ITA, além de algumas universidades que mantém convênio com as mesmas, como a COPPE/UFRJ e a USP, que possuem alunos de pós-graduação originários das FFAA.

Todavia, em se considerando a comunidade internacional, considera-se o interesse sobre o tema bastante pronunciado apesar do mesmo possuir um grande interesse estratégico e de ser tratado de forma sigilosa. Existem inúmeros eventos, periódicos e livros-texto que podem ser encontrados sobre o assunto, sendo que alguns eventos significativos onde se apresenta o estado da arte, reunindo autores bastante seletos, são organizados pelo *IEEE - Signal Processing Society.*

A monitoração de contatos ou múltiplos contatos existe não somente em aplicações militares, mas também em aplicações civis [26]. Um exemplo seria o emprego de algoritmos estimadores em sistemas de rastreamento de veículos automotores através de sistemas de GPS – *Global Positioning System* ou em sistemas de localização de objetos utilizando tecnologia *wireless*.

Outras aplicações que utilizam algoritmos de estimação são: controle de processos em plantas químicas, onde os sensores envolvidos realizam a medida de pressão, temperatura, vazão e análise de gases; predição de chuva em sistemas de controle de nível de rios para prevenção de alagamentos e controle de barragens, onde as entradas são o nível de água, o índice pluviométrico e os sinais provenientes de radares meteorológicos; e sistemas inerciais de navegação de aviões, de satélites e de naves espaciais, que empregam medidas provenientes de sistemas de imagem, de sensores do tipo giroscópio e de sensores do tipo acelerômetro.

Além disso, os países como maior desenvolvimento industrial e comercial no segmento de armamentos, como os Estados Unidos da América, a França, a Inglaterra, a Alemanha, o Canadá, a Índia e o Japão, mantêm e custeiam centros de pesquisas e linhas de pesquisas de universidades públicas e privadas para que sejam desenvolvidos trabalhos que permitam que os mesmos estejam próximos do estado da arte.

4. Proposição da Tese

Além do estudo teórico sobre técnicas e algoritmos utilizados em problemas tradicionalmente conhecidos como *Target Motion Analysis* (TMA) publicados em trabalhos anteriores, e que foi apresentado no item 3, esta tese propõe-se realizar a análise do movimento de veículos utilizando técnicas de inteligência artificial. Para alcançar este resultado, os seguintes tópicos serão tratados: proposição e desenvolvimento de um simulador de movimentos de contatos; avaliação das técnicas e dos algoritmos inicialmente estudados; e, por fim, conforme foi proposto, implementação e avaliação de um novo método que utiliza conceitos e técnicas de inteligência artificial, a fim de estimar e predizer a posição e a velocidade de veículos navais, de superfície e submarinos.

Os resultados obtidos pelo novo método serão comparados com os obtidos pelas técnicas e pelos algoritmos inicialmente estudados, a fim de validar esta proposta.

4.1. Objetivo Principal

O objetivo principal desta Tese é propor um método e o desenvolvimento de uma ferramenta computacional que utiliza técnicas de Inteligência Artificial para avaliar e estimar (predizer) a posição e a velocidade de veículos navais, de superfície e submarinos. Os cálculos e estimativas são efetuados sobre uma base de dados, contendo valores reais e/ou simulados. Ao término do trabalho, pretende-se demonstrar que a ferramenta desenvolvida é capaz de apresentar desempenho, medido em termos de precisão e tempo para obtenção do resultado, compatível com o estado da arte.

A motivação em desenvolver esta Tese é a de que as idéias aqui desenvolvidas possam ser empregadas em aplicações reais utilizando a tecnologia disponível e dominada pelo Brasil, uma vez que a total compreensão dos algoritmos de acompanhamento implementados pelos fabricantes dos sistemas operativos em utilização atualmente, no que diz respeito ao seu funcionamento, não é possível apenas com a documentação existente.

Partindo dos métodos e algoritmos pesquisados na revisão bibliográfica, foi desenvolvido um método híbrido que permite estimar a posição, a velocidade e o rumo

de um contato designado, tendo como dados apenas as marcações do contato e as posições do navio observador.

O método proposto utiliza o Estimador Pseudolinear com uma modificação que otimiza o processo de busca pela melhor solução da posição estimada do contato. O Estimador Pseudolinear Modificado (PL-MOD) utiliza o conceito de elitismo, isto é, guardar o resultado com melhor aptidão e somente alterar o resultado obtido na iteração corrente caso este possua aptidão maior do que a encontrada até então. Este conceito existe nas técnicas de inteligência artificial, como as Técnicas Evolutivas (Algoritmos Genéticos) e Sócio-Cognitivas (*Particle Swarm*) [17,18,19,20,21]. Em trabalhos relacionados com engenharia eletrônica [8,9] e com análise de agrupamentos [7], estas técnicas foram utilizadas e os resultados alcançados confirmaram que as mesmas são mais eficientes que os métodos que empregam a busca exaustiva.

No presente trabalho, não foram implementados outros conceitos existentes nestas técnicas, mas futuramente poderiam ser implementadas outras variantes das mesmas em partes específicas do algoritmo, conforme será relatado no item "Trabalhos Futuros".

O processo de análise dos resultados utiliza métricas de desempenho adequadas para realizar a comparação dos resultados alcançados pelas técnicas pesquisadas com os resultados obtidos pela técnica proposta.

Como em qualquer problema de engenharia, não existe apenas uma solução heurística na resolução do mesmo, mas sim várias soluções, que dependem, *a priori*, dos pressupostos que foram pré-estabelecidos. Portanto, faz-se necessário estabelecer os mesmos, para que o desenvolvimento do presente trabalho ocorra de maneira contínua, de modo a conseguir ao seu final, responder as questões indagadas inicialmente.

4.2. Pressupostos Empregados na Proposta de Solução do Problema

4.2.1. Simulador do Movimento do Contato e do Navio Observador

Para gerar os conjuntos de dados utilizados ns testes dos algoritmos descritos nos trabalhos estudados, foi projetado um simulador de trajetórias do contato e do navio observador.

O mesmo permite não somente recriar as situações propostas anteriormente como também simular novas cinemáticas de movimento para o contato e para o navio observador.

As posições reais do contato e do navio observador, bem como as marcações do contato tomadas pelo navio observador, em qualquer iteração podem ser obtidas com o uso do simulador que calcula as mesmas a partir das condições iniciais e das equações do Movimento Retilíneo Uniforme (MRU).

A figura 4.1 abaixo representa uma situação típica de cinemática de movimento de um navio observador e de um contato em um cenário marítimo. As variáveis e os parâmetros existentes, bem como as incógnitas do problema, estão nela representados.

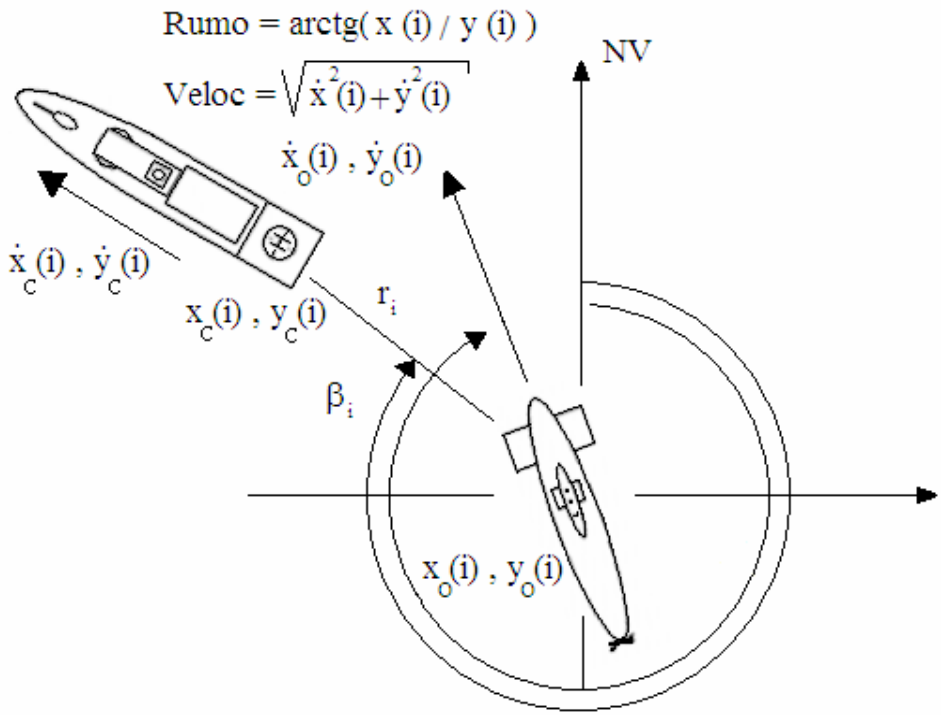


Figura 4.1: Situação típica de cinemática de movimento de navios.

Considerando que $(x_C(0), y_C(0))$ é a posição inicial do contato no instante de tempo zero (0), que $(x_O(0), y_O(0))$ é a posição inicial do navio observador no mesmo instante de tempo e assumindo que os mesmos estão se movendo com velocidade constante $(\dot{x}_C, \dot{y}_C)$ e $(\dot{x}_O, \dot{y}_O)$, respectivamente, a posição do contato e do navio observador no instante de tempo iT onde T é o período de observação e i é o índice de tempo, são dadas pelas seguintes equações do MRU:

$$x_C(i) = x_C(0) + iT \cdot \dot{x}_C \tag{4.1}$$

$$y_C(i) = y_C(0) + iT \cdot \dot{y}_C \tag{4.2}$$

$$x_O(i) = x_O(0) + iT \cdot \dot{x}_O \tag{4.3}$$

$$y_O(i) = y_O(0) + iT \cdot \dot{y}_O \tag{4.4}$$

Conforme pode ser observado da figura 4.1, as marcações verdadeiras para o contato medidas pelo observador com relação ao eixo y (NV) são obtidas a partir do arco-tangente da marcação instantânea, conforme a equação abaixo:

$$\overline{\beta}_i = arctg\left(\frac{x_C(i) - x_O(i)}{y_C(i) - y_O(i)}\right) \tag{4.5}$$

Portanto, como os algoritmos a serem testados necessitam das sucessivas posições do observador $(x_O(i), y_O(i))$ e da série de medidas das marcações para o contato $\overline{\beta}_i$, onde i = 0, 1, 2, ..., *k*, são obtidos vetores que contem os resultados das equações acima, como dados de saída do simulador.

4.2.2. Pressupostos do Movimento do Contato e do Navio Observador

Pressupõe-se que o contato desloca-se com rumo e velocidade constantes e que os mesmos não sofrem alterações significativas durante o intervalo de observação utilizado no cálculo da estimativa de sua posição. Todos os trabalhos que foram estudados na revisão bibliográfica fazem este tipo de restrição, exceto o que trata o problema utilizando os *Filtros de Kalman* e suas variantes [26]. No entanto, o mesmo utiliza diversas plataformas fixas de aquisição dos dados de marcação. A inserção de ruído aos dados originais pode mostrar o quão robusto são os algoritmos às variações de velocidade e rumo do contato, sendo que esta análise será realizada no capítulo 6.

Considera-se também que em cada pernada do navio observador, tanto na superfície da água quanto abaixo da mesma, caso seja um submarino, o rumo e velocidade do mesmo não sofrem variações significativas.

Para efeito de estudo nesta dissertação, uma corrida do navio observador, será formada por duas pernadas, dentro de um intervalo de tempo de observação, e neste caso, os rumos e velocidades de cada pernada poderão sofrer variações, mas somente entre as pernadas, em conformidade com o parágrafo anterior.

Normalmente, ao se considerar que o contato desloca-se com velocidade constante, sua trajetória pode ser aproximada por arcos de circunferências. Deste modo, a distância do navio observador ao contato é muito maior do que a distancia percorrida pelo navio observador durante o intervalo de medição.

Nos problemas de navegação, tanto aérea, quanto marítima, as distâncias são medidas em milhas náuticas (NM), sendo que uma NM é igual a 1852 metros (m).

As velocidades são expressas, portanto na unidade nós, que equivale a uma NM por hora, ou seja, uma NM/h é igual a um nó.

Nos problemas que envolvem a resolução do problema do tiro naval, as distâncias são medidas jardas (jds), que possui a seguinte relação com a NM: Uma milha náutica (NM) é igual a 2025,4 jardas (jds).

4.2.3. Métrica de erro adotada – Raiz quadrada do erro médio quadrático (RMSE)

O mesmo é freqüentemente utilizado para medir a diferença entre dois valores, sendo um deles o valor real, e o outro o valor predito ou estimado, através de um algoritmo ou modelo. Esta diferença é freqüentemente chamada de resíduo.

No caso do problema em questão, deseja-se saber, com relação às posições reais simuladas $x_C(k), y_C(k)$ ocupadas pelo contato durante o intervalo de medição, e que foram obtidas a partir das equações do MRU, considerando-se as condições iniciais do contato, o quão erradas em termos de distância foram as estimativas de posição $\bar{x}_C(k), \bar{y}_C(k)$ obtidas pelos algoritmos estimadores.

A expressão matemática da medida de erro utilizada nesta dissertação está abaixo representada:

$$RMSE = \sqrt{\left(\bar{x}_C(k) - x_C(k)\right)^2 + \left(\bar{y}_C(k) - y_C(k)\right)^2} \tag{4.6}$$

4.2.4. Emprego dos Algoritmos de Estimação Pseudolinear

Os algoritmos pseudolineares [5], como já visto, são alguns dos métodos matemáticos utilizados em problemas em que se deseja estimar o valor de uma variável, tendo-se conhecimento somente de alguns poucos dados disponíveis, com relação ao problema em estudo.

Os mesmos foram escolhidos devido à particularidade do problema existente, onde se deseja conhecer a posição futura de um contato, tendo apenas conhecimento de uma série de medidas de marcações do contato tomadas pelos sensores do navio observador, e das posições sucessivas deste navio durante o intervalo de medições.

Os outros algoritmos estudados como os filtros de *Kalman*, e suas variantes EKF e UKF, normalmente utilizam também como entrada inicial a distância do contato ao navio observador. Uma variante dos mesmos [26] foi modelada para trabalhar em problemas de B-O, mas necessita de duas ou mais plataformas fixas de medição das marcações do contato, interligadas através de um *link* de dados com transmissão em tempo real, de forma a permitir estimar a posição do contato. Deste modo, consegue resolver o problema sem a necessidade de uma plataforma de sensoriamento realizar uma corrida com no mínimo duas pernadas com rumos e velocidades diferentes, conforme já estudado em outros métodos.

Já os algoritmos que são implementados utilizando os sistemas *fuzzy* [2,3,4], empregam um número maior de parâmetros de entrada, com relação aos sistemas de sensores existentes e disponíveis para obtenção dos dados.

Fora isso o método da distância de EKELUND [1], foi implementado como método alternativo de modo a permitir realizar a comparação dos resultados obtidos.

As técnicas utilizadas na presente tese são algumas dentre as várias, das mais recentes ferramentas empregadas com o objetivo de buscar uma solução sem o uso de técnicas de busca exaustiva, também conhecida por “força bruta”.

A técnica proposta é mais indicada do que a busca exaustiva implementada com o método PL original [5] devido ao fato de que esta última, além de necessitar de um esforço computacional enorme, perde desempenho à medida que a solução do problema converge para outro mínimo local, deixando a solução de melhor resultado perdida.

Assim, o PL-MOD foi derivado das técnicas de inteligência artificial e permite uma busca otimizada e robusta, utilizando uma capacidade computacional menor.

4.2.4.1 Modelagem Matemática do Método Pseudolinear Original

Um problema típico de determinação da posição ocupada por um contato em um cenário marítimo está representado na figura 4.2. As variáveis e os parâmetros existentes, bem como as incógnitas do problema, estão nela representados.

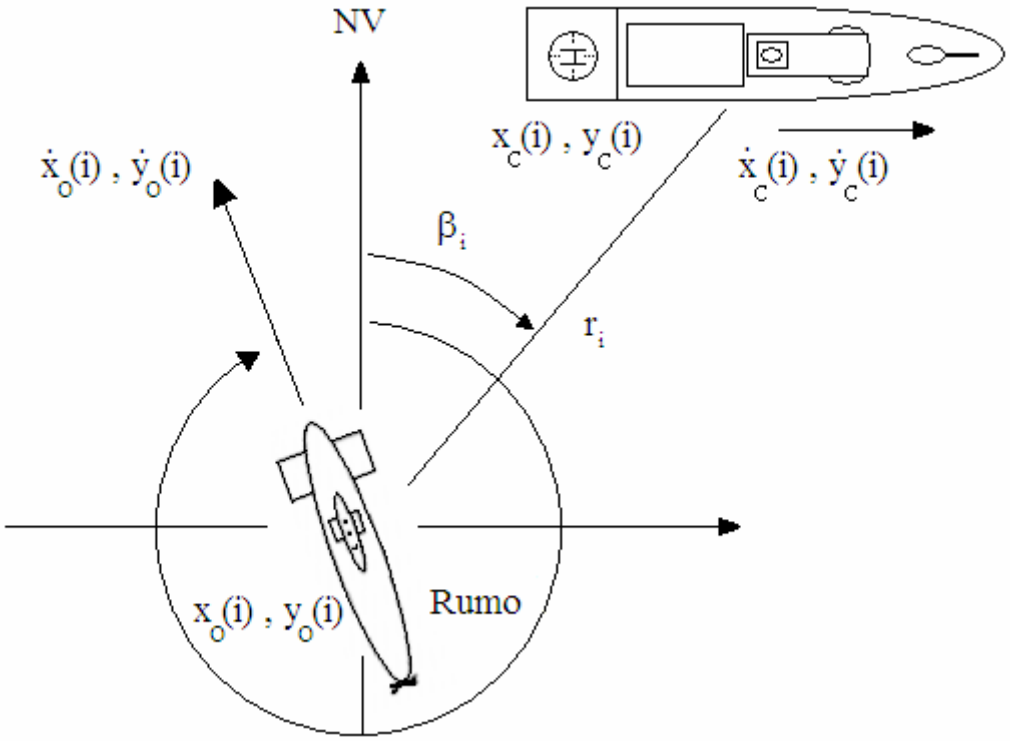


Figura 4.2: Posições, Marcações, Distâncias, Rumos e Velocidades dos Navios - Método Pseudolinear Original.

Considerando que $(x_C(0), y_C(0))$ é a posição inicial do contato no instante de tempo zero (0) e assumindo que o mesmo está se movendo com velocidade constante $(\dot{x}_C, \dot{y}_C)$, a posição do contato no instante de tempo iT onde T é o período de observação e i é o índice de tempo, são dadas pelas equações 4.1 e 4.2 do MRU, que foram vistas anteriormente.

Para determinar a trajetória estimada do contato, o navio observador mede as marcações para o mesmo com respeito ao eixo y (NV):

$$\beta_i = \overline{\beta}_i + e_{\beta,i} \tag{4.7}$$

onde $\overline{\beta}_i$ é a marcação verdadeira e $e_{\beta,i}$ é a componente independente do ruído de medida identicamente distribuído com densidade de potência σ_β^2.

Sendo conhecidas as sucessivas posições do observador ($x_O(i), y_O(i)$) e a série de medidas das marcações para o contato β_i , onde i = 0, 1, 2, ..., *k*, deseja-se obter a posição estimada do mesmo para cada instante de tempo *k*, ou seja, ($x_C(k), y_C(k)$).

Considerando que o contato está se movendo com velocidade constante, o problema de estimar as posições do mesmo fica reduzido a calcular a sua posição inicial ($x_C(0), y_C(0)$) e a sua velocidade ($\dot{x}_C, \dot{y}_C$).

Conforme pode ser observado da figura 4.2, as marcações verdadeiras para o contato medidas pelo observador possuem relação direta com as coordenadas das posições dos navios através da tangente da marcação instantânea, conforme a equação abaixo:

$$\tan(\overline{\beta}_i) = \frac{sen(\overline{\beta}_i)}{\cos(\overline{\beta}_i)} = \frac{x_C(i) - x_O(i)}{y_C(i) - y_O(i)} \tag{4.8}$$

Realizando a multiplicação cruzada, a equação acima fica reduzida a:

$$\cos(\overline{\beta}_i)[x_C(i) - x_O(i)] - sen(\overline{\beta}_i)[y_C(i) - y_O(i)] = 0 \tag{4.9}$$

No entanto, as marcações verdadeiras $\overline{\beta}_i$ não são conhecidas e somente as medidas β_i que contém ruído estão disponíveis. A substituição de $\overline{\beta}_i$ por β_i na equação (4.9) dá como resultado uma equação de erro pseudolinear, conforme pode ser visto abaixo:

$$e_\beta(i) = \cos(\beta_i)[x_C(i) - x_O(i)] - sen(\beta_i)[y_C(i) - y_O(i)] \tag{4.10}$$

Onde $e_\beta(i) = -r_i \tan(\beta_i)$ e r_i é a distância do contato ao observador no instante de tempo *i*.

Dado que o vetor de parâmetros desconhecidos do contato seja:

$$\boldsymbol{\mu}_\beta = \left[x_T(0), \dot{x}_C, y_C(0), \dot{y}_C \right]^T \tag{4.11}$$

que o vetor de medida é dado por:

$$\mathbf{g}_\beta = \left[\cos(\beta_0) x_O(0) - sen(\beta_0) y_O(0), ..., \cos(\beta_k) x_O(k) - sen(\beta_k) y_O(k) \right]^T \tag{4.12}$$

e finalmente que a matriz de medidas é dada por:

$$\mathbf{A}_\beta = \begin{bmatrix} \cos(\beta_0) & 0 & -sen(\beta_0) & 0 \\ \cos(\beta_1) & T\cos(\beta_1) & -sen(\beta_1) & -Tsen(\beta_1) \\ \vdots & \vdots & \vdots & \vdots \\ \cos(\beta_k) & kT\cos(\beta_k) & -sen(\beta_k) & -kTsen(\beta_k) \end{bmatrix} \tag{4.13}$$

Substituindo (4.1) e (4.2) em (4.10) e amostrando os erros da equação como um vetor $\mathbf{e}_\beta = \left[e_\beta(0), e_\beta(1), e_\beta(2), ..., e_\beta(k),\right]^T$ obtém-se a seguinte equação da algébrica:

$$\mathbf{e}_\beta = \mathbf{A}_\beta \boldsymbol{\mu}_\beta - \mathbf{g}_\beta \tag{4.14}$$

Minimizando $\mathbf{e}_\beta^T \mathbf{e}_\beta$ com relação a $\boldsymbol{\mu}_\beta$ obtemos a solução de mínimos quadrados com restrição (CLS):

$$\hat{\boldsymbol{\mu}}_\beta = (\mathbf{A}_\beta^T \mathbf{A}_\beta)^{-1} \mathbf{A}_\beta^T \mathbf{g}_\beta \tag{4.15}$$

Esta solução de CLS é parcial. Colocando $\mathbf{g}_\beta = \mathbf{A}_\beta \boldsymbol{\mu}_\beta^0 - \bar{\mathbf{e}}_\beta$ na equação (4.14) e tomando a expectativa de rendimento, obtém-se:

$$E\left[\hat{\boldsymbol{\mu}}_\beta\right] = \boldsymbol{\mu}_\beta^0 - E\left[(\mathbf{A}_\beta^T \mathbf{A}_\beta)^{-1} \mathbf{A}_\beta^T \bar{\mathbf{e}}_\beta\right] \tag{4.16}$$

onde $\boldsymbol{\mu}_\beta^0$ é a solução verdadeira e $\bar{\mathbf{e}}_\beta$ é o vetor de erro de equação avaliada em $\boldsymbol{\mu}_\beta^0$.

Nota-se que as medidas de marcações com ruído aparecem tanto em $\mathbf{A}_\beta$ como em $\bar{\mathbf{e}}_\beta$, e conseqüentemente $E\left[(\mathbf{A}_\beta^T \mathbf{A}_\beta)^{-1} \mathbf{A}_\beta^T \bar{\mathbf{e}}_\beta\right] \neq E\left[(\mathbf{A}_\beta^T \mathbf{A}_\beta)^{-1} \mathbf{A}_\beta^T\right] \cdot E\left[\bar{\mathbf{e}}_\beta\right]$.

Como resultado, o segundo termo à direita de (4.16) não é igual a zero o que gera solução parcial, também conhecida por *bias* ou nível de *off-set*. Este nível pode ser muito significante. Pode não diminuir com o aumento do número de amostras de marcações, e aumentar ainda mais com o aumento do ruído presente nas medidas.

O método proposto [5] gera uma solução assintoticamente imparcial, ou seja, sem *bias*. Este método define uma matriz de medida aumentada:

$$\mathbf{A}_{\mu,\beta} = \left[\mathbf{A}_\beta, -\mathbf{g}_\beta\right] \tag{4.17}$$

e um vetor de solução aumentado:

$$\boldsymbol{\theta}_\beta = h_\beta \left[\boldsymbol{\mu}_\beta^T, 1\right]^T \tag{4.18}$$

onde h_β é uma constante escalar cuja função será discutida depois da equação (4.25) e antes de (4.28). A fim de desenvolver a formulação proposta, assume-se que os erros de medida de marcação são pequenos, como na formulação do método PL e no

método da variável instrumental (*Instrumental Variable* – IV). Se os erros forem grandes, o efeito de *thresholding* ocorrerá uma vez que a precisão de estimação diverge nitidamente do método *Cramer-Rao Lower Bound* (CRLB). Este é um fenômeno comum por usar um estimator linear para resolver um problema de estimação não-linear.

Considerando as condições de medida com ruído pequeno, temos que $\cos(e_{\beta,i}) \cong 1$ e $sen(e_{\beta,i}) \cong e_{\beta,i}$. Aplicando a relação trigonométrica de soma de senos e co-senos e substituindo nas mesmas a equação (4.7), temos que:

$$\cos(\beta_i) = \cos(\overline{\beta}_i) \cdot \cos(e_{\beta,i}) - sen(\overline{\beta}_i) \cdot sen(e_{\beta,i})$$

$$\cos(\beta_i) \cong \cos(\overline{\beta}_i) - e_{\beta,i} \cdot sen(\overline{\beta}_i) \tag{4.19}$$

$$sen(\beta_i) = sen(\overline{\beta}_i) \cdot \cos(e_{\beta,i}) + \cos(\overline{\beta}_i) \cdot sen(e_{\beta,i})$$

$$sen(\beta_i) \cong sen(\overline{\beta}_i) + e_{\beta,i} \cdot \cos(\overline{\beta}_i) \tag{4.20}$$

Realizando a substituição dos mesmos nos elementos de $\mathbf{g}_\beta$ e $\mathbf{A}_\beta$ em (4.12) e (4.13), obtém-se:

$$\mathbf{A}_{\mu,\beta} = \overline{\mathbf{A}}_{\mu,\beta} + \widetilde{\mathbf{A}}_{\mu,\beta} \tag{4.21}$$

onde $\overline{\mathbf{A}}_{\mu,\beta}$ é obtida substituindo-se as marcações com ruído β_i pelas marcações verdadeiras (sem ruído) $\overline{\beta}_i$ na matriz $\mathbf{A}_{\mu,\beta}$, e

$$\widetilde{\mathbf{A}}_{\mu,\beta} = diag\{e_{\beta,0}, e_{\beta,1}, e_{\beta,2}, ..., e_{\beta,k}\} \times \left[\overline{\boldsymbol{\mu}}_{\beta,0}, \overline{\boldsymbol{\mu}}_{\beta,1}, \overline{\boldsymbol{\mu}}_{\beta,2}, ..., \overline{\boldsymbol{\mu}}_{\beta,k}\right]^T \tag{4.22}$$

onde

$$\overline{\boldsymbol{\mu}}_{\beta,i} = \Big[-sen(\overline{\beta}_i), -iTsen(\overline{\beta}_i), -\cos(\overline{\beta}_i), -iT\cos(\overline{\beta}_i), \tag{4.23}$$

$$x_O(i) \cdot sen(\overline{\beta}_i) + y_O(i) \cdot \cos(\overline{\beta}_i) \Big]^T$$

$\overline{\mathbf{A}}_{\mu,\beta}$ é o componente da matriz aumentada devido ao ruído.

Definindo a matriz restrita (limitada) $\mathbf{W}_\beta$ como:

$$\mathbf{W}_\beta = \sum_{i=0}^{k} \overline{\boldsymbol{\mu}}_{\beta,i} \cdot \overline{\boldsymbol{\mu}}_{\beta,i}^T \tag{4.24}$$

Nota-se que de (4.22) decorre a seguinte equação $E\left[\widetilde{\mathbf{A}}_{\mu,\beta}^{T}\cdot\widetilde{\mathbf{A}}_{\mu,\beta}\right]=\sigma_{\beta}^{2}\cdot\mathbf{W}_{\beta}$. Conseqüentemente $\mathbf{W}_{\beta}$ representa a matriz de correlação da componente de ruído em $\mathbf{A}_{\mu,\beta}$, e seu papel será descrito na seqüência. O vetor de erro da equação (4.14) pode ser reescrito em termos de $\mathbf{A}_{\mu,\beta}$ e $\boldsymbol{\theta}_{\beta}$ sendo então definido nas equações (4.17) e (4.18) como $\mathbf{e}_{\beta}=\mathbf{A}_{\mu,\beta}\cdot\boldsymbol{\theta}_{\beta}/h_{\beta}$.

O método proposto forma a matriz de correlação aumentada $\mathbf{R}_{\beta}=\mathbf{A}_{\mu,\beta}^{T}\cdot\mathbf{A}_{\mu,\beta}$ e achando $\boldsymbol{\theta}_{\beta}$ através da minimização de $\boldsymbol{\theta}_{\beta}^{T}\mathbf{R}_{\beta}\boldsymbol{\theta}_{\beta}$ chega-se a equação quadrática abaixo:

$$\boldsymbol{\theta}_{\beta}^{T}\mathbf{W}_{\beta}\boldsymbol{\theta}_{\beta}=1 \tag{4.25}$$

A constante unitária (1) na restrição é arbitrária. Usando um valor diferente de da unidade simplesmente efetua-se a mudança da constante h_{β} na equação de $\boldsymbol{\theta}_{\beta}$ (4.18). A minimização restrita é necessária aqui porque foi ampliado de quatro, em $\boldsymbol{\mu}_{\beta}$, para cinco, em $\boldsymbol{\theta}_{\beta}$, o tamanho do vetor de parâmetros desconhecidos. A restrição reduz em um grau de liberdade a dimensão do espaço de busca, de forma que o número de parâmetros livres permanece em quatro.

De forma a justificar que a equação (4.25) gera uma solução boa, será feita a suposição de que o ruído do processo de medida está ausente, ou seja, é nulo. Neste caso, $\mathbf{R}_{\beta}=\mathbf{A}_{\mu,\beta}^{T}\cdot\mathbf{A}_{\mu,\beta}$. É fácil verificar, através da equação (4.9), que isso ocorre se $\boldsymbol{\theta}_{\beta}^{T}\mathbf{R}_{\beta}\boldsymbol{\theta}_{\beta}$ atinge seu valor mínimo igual a zero quando a solução verdadeira é alcançada.

A restrição $\boldsymbol{\theta}_{\beta}^{T}\mathbf{W}_{\beta}\boldsymbol{\theta}_{\beta}=1$ evita a solução trivial e está satisfeito quando h_{β} é igual a $\left(\left[\boldsymbol{\mu}_{\beta}^{o\,T},1\right]\cdot\mathbf{W}_{\beta}\cdot\left[\boldsymbol{\mu}_{\beta}^{o\,T},1\right]^{T}\right)^{-1/2}$, onde $\boldsymbol{\mu}_{\beta}^{o}$ representa a solução verdadeira de $\boldsymbol{\mu}_{\beta}$.

Na presença de ruído, conforme pode ser visto na equação (4.21), $\mathbf{R}_{\beta}\cong\overline{\mathbf{A}}_{\mu,\beta}^{T}\cdot\overline{\mathbf{A}}_{\mu,\beta}+\sigma_{\beta}^{2}\cdot\mathbf{W}_{\beta}$ quando o número de medidas é grande. Sob a restrição, a componente de ruído em $\boldsymbol{\theta}_{\beta}^{T}\mathbf{R}_{\beta}\boldsymbol{\theta}_{\beta}$ não afeta a minimização de $\boldsymbol{\theta}_{\beta}^{T}\mathbf{R}_{\beta}\boldsymbol{\theta}_{\beta}$, de forma que também produz a solução verdadeira. A restrição neste caso não só evita a solução trivial, mas também restringe o espaço de procura em $\boldsymbol{\theta}_{\beta}$ dentro de uma hiperesfera definida pela restrição de forma que a componente de ruído em $\mathbf{R}_{\beta}$ não afeta a minimização.

O método do multiplicador de Lagrange [6] pode resolver o problema da minimização restrita através do uso de uma função de custo auxiliar:

$$\xi = \boldsymbol{\theta}_\beta^T \mathbf{R}_\beta \boldsymbol{\theta}_\beta + \lambda\left(1 - \boldsymbol{\theta}_\beta^T \mathbf{W}_\beta \boldsymbol{\theta}_\beta\right) \tag{4.26}$$

onde λ é o multiplicador de Lagrange. Tirando as derivadas parciais de ξ com relação à $\boldsymbol{\theta}_\beta$ resulta na equação abaixo:

$$\mathbf{R}_\beta \boldsymbol{\theta}_\beta = \lambda \mathbf{W}_\beta \boldsymbol{\theta}_\beta \tag{4.27}$$

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{B}\mathbf{x}$$

que indica que $\boldsymbol{\theta}_\beta$ é o autovetor generalizado. Realizando previamente a multiplicação da equação (4.27) por $\boldsymbol{\theta}_\beta^T$ resulta em $\lambda = \boldsymbol{\theta}_\beta^T \mathbf{R}_\beta \boldsymbol{\theta}_\beta$ que é a quantidade a ser minimizada.

Conseqüentemente $\boldsymbol{\theta}_\beta$ é o autovetor generalizado do par $\left(\mathbf{R}_\beta, \mathbf{W}_\beta\right)$ que é obtido a partir do menor autovalor generalizado associado. Uma vez que $\boldsymbol{\theta}_\beta$ foi encontrado, então $h_\beta = \theta_\beta(5)$ e portanto o vetor de parâmetros do contato é calculado através da equação abaixo:

$$\boldsymbol{\mu}_\beta = \frac{\left[\theta_\beta(1), \theta_\beta(2), \theta_\beta(3), \theta_\beta(4)\right]^T}{\theta_\beta(5)} = \left[x_C(0), \dot{x}_C, y_C(0), \dot{y}_C\right]^T \tag{4.28}$$

A solução proposta necessita da matriz restrita $\mathbf{W}_\beta$ que foi apresentada pela equação (4.24). Esta matriz é dependente das marcações verdadeiras que não são conhecidas. As mesmas podem ser substituídas pelas marcações com ruído obtidas dos sensores do navio, como o sonar e o radar, sendo que os erros resultantes são desprezíveis quando o ruído for desprezível.

Na realidade, pode-se derivar a equação (4.23) - $\overline{\boldsymbol{\mu}}_{\beta,i} = \left[-sen(\overline{\beta}_i), -iTsen(\overline{\beta}_i), -\cos(\overline{\beta}_i), -iT\cos(\overline{\beta}_i),\ x_O(i)\cdot sen(\overline{\beta}_i) + y_O(i)\cdot\cos(\overline{\beta}_i)\right]^T$, utilizando as equações (4.19) - $[\cos(\beta_i) \cong \cos(\overline{\beta}_i) - e_{\beta,i}\cdot sen(\overline{\beta}_i)]$ e (4.20) - $[sen(\beta_i) \cong sen(\overline{\beta}_i) + e_{\beta,i}\cdot\cos(\overline{\beta}_i)]$ e expressando $\overline{\beta}_i$ como $\beta_i - e_{\beta,i}$, resultando na equação a seguir:

$$\overline{\boldsymbol{\mu}}_{\beta,i} = \Big[-sen(\beta_i), -iTsen(\beta_i), -\cos(\beta_i), -iT\cos(\beta_i), \tag{4.29}$$

$$x_O(i)\cdot sen(\beta_i) + y_O(i)\cdot\cos(\beta_i)\Big]^T +$$

$$e_{\beta,i}\cdot\Big[\cos(\beta_i), iT\cos(\beta_i), -sen(\beta_i), -iTsen(\beta_i),$$

$$-x_O(i)\cdot\cos(\beta_i) + y_O(i)\cdot sen(\beta_i)\Big]^T$$

$$\overline{\boldsymbol{\mu}}_{\beta,i} = \boldsymbol{\mu}_{\beta,i} + e_{\beta,i}\cdot\mathbf{v}_{\beta,i}$$

Conseqüentemente quando $e_{\beta,i}$ for pequeno, $\overline{\boldsymbol{\mu}}_{\beta,i}$ pode ser aproximado usando as marcações com ruído. Por outro lado, a equação (4.29) pode ser usada para gerar a matriz $\mathbf{W}_\beta$, como abaixo representada:

$$\mathbf{W}_\beta = \sum_{i=0}^{k}\left(\boldsymbol{\mu}_{\beta,i}\cdot\boldsymbol{\mu}_{\beta,i}^T - \sigma_\beta^2\cdot\mathbf{v}_{\beta,i}\cdot\mathbf{v}_{\beta,i}^T\right) \tag{4.30}$$

Outra alternativa para contornar a situação de ruído alto é usar as medidas com ruído dentro da matriz $\mathbf{W}_\beta$, obtendo uma estimativa inicial. Em seguida, atualiza-se $\mathbf{W}_\beta$ de modo a obter uma solução mais precisa.

O algoritmo de estimação proposto pode ser assim resumido:

1. A matriz de correlação aumentada $\mathbf{R}_\beta = \mathbf{A}_{\mu,\beta}^T\cdot\mathbf{A}_{\mu,\beta}$ é construída, sendo que a matriz $\mathbf{A}_{\mu,\beta}$ foi definida pela equação (4.17). Na matriz $\mathbf{W}_\beta$, que foi definida pela equação (4.24), as marcações verdadeiras são substituídas pelas medidas com ruído disponíveis nos sensores de bordo;
2. Faz-se a decomposição generalizada de autovalores do par de matrizes $\left(\mathbf{R}_\beta, \mathbf{W}_\beta\right)$;
3. Seleciona-se o autovetor generalizado $\boldsymbol{\theta}_\beta$ associado ao menor autovalor obtido; e
4. A partir de $\boldsymbol{\mu}_\beta = \dfrac{\left[\theta_\beta(1), \theta_\beta(2), \theta_\beta(3), \theta_\beta(4)\right]^T}{\theta_\beta(5)} = \left[x_C(0), \dot{x}_C, y_C(0), \dot{y}_C\right]^T$, que é a equação (4.28), obtém-se finalmente a posição inicial designada e vetor de velocidade do contato.

4.2.4.2 Algoritmo estimador PL Original para o problema BO-TMA

Os dados necessários para realizar à estimativa da posição inicial do contato e seu vetor de velocidade são as posições conhecidas do navio observador e as marcações conhecidas do contato que foram medidas pelo navio observador. As posições são normalmente fornecidas pela agulha giroscópica do navio e as marcações do contato são disponibilizadas pelos sistemas de sensores do mesmo, mais comumente pelo equipamento de sonar ou de MAGE.

A principio, não é necessário que se disponha previamente de qualquer tipo de conhecimento sobre as condições iniciais do movimento do contato designado. O algoritmo utiliza uma parcela inicial dos dados amostrados para realizar a sua inicialização. Após esta inicialização, o mesmo realiza a estimativa das incógnitas desejadas.

Na tabela 4.1, está representado um extrato resumido do Algoritmo Estimador PL Original para um problema de BO-TMA. O mesmo pode ser implementado em diversos tipos de linguagens de programação, sendo apenas necessário que se opere com cálculo de matrizes e se resolva o problema de autovalores e autovetores generalizados.

O valor de k_0 no trabalho original é igual a 50, o que corresponde a 10% da massa de dados a ser simulada. O valor da constante η é da ordem de 10^{-10} para interromper o número de iterações na busca da solução no *loop* mais interno.

Foi necessário também acrescentar um critério de parada quando o algoritmo não apresenta a convergência esperada ao se alcançar um certo número de iterações. Nas primeiras vezes em que o algoritmo original foi testado, o mesmo ficava eternamente "preso" neste *loop* interno, sendo necessário forçar sua parada. A modificação implementada está destacada em fundo amarelo, para melhor visualização.

Algoritmo estimador para o problema BO-TMA

1. Dados necessários para a execução do algoritmo estimador PL BO-TMA:

$(x_0(i), y_0(i))$, i = 0, 1, 2, ..., Posições conhecidas do navio observador.

β_i, i = 0, 1, 2, ..., Marcações conhecidas do contato.

2. Inicialização do algoritmo utilizando k_0 + 1 medidas.

$\mathbf{r}_{\beta,i} = \left[\cos(\beta_i), iT\cos(\beta_i), -sen(\beta_i), -iTsen(\beta_i), (-\cos(\beta_i)\cdot x_0(i) + sen(\beta_i)\cdot y_0(i))\right]^T$, i = 0, 1, 2, ..., k_0

$\boldsymbol{\mu}_{\beta,i} = \left[-sen(\beta_i), -iTsen(\beta_i), -\cos(\beta_i), -iT\cos(\beta_i), (sen(\beta_i)\cdot x_0(i) + \cos(\beta_i)\cdot y_0(i))\right]^T$, i = 0, 1, 2, ..., k_0

$\mathbf{A}_{\mu,\beta} = \left[\mathbf{r}_{\beta,0}, \mathbf{r}_{\beta,1}, \mathbf{r}_{\beta,2}, \mathbf{r}_{\beta,3}, ..., \mathbf{r}_{\beta,k_0}\right]^T$

$\mathbf{R}_{\beta,k_0} = \mathbf{A}^T_{\mu,\beta} \cdot \mathbf{A}_{\mu,\beta}$

$\mathbf{W}_{\beta,k_0} = \sum_{i=0}^{k_0} \boldsymbol{\mu}_{\beta,i} \cdot \boldsymbol{\mu}^T_{\beta,i}$

$\boldsymbol{\theta}_{\beta,i} =$ Autovetor Generalizado correspondente ao menor Autovalor obtido de eig($\mathbf{R}_{\beta,k_0}$, $\mathbf{W}_{\beta,k_0}$)

$\mathbf{F}_{\beta,k_0} = \mathbf{R}^{-1}_{\beta,k_0}$

3. *Loop* .principal onde se obtém as Estimativas

for k = k0 +1 , k0 +2 , ...

$\mathbf{r}_{\beta,k} = \left[\cos(\beta_k), kT\cos(\beta_k), -sen(\beta_k), -kTsen(\beta_k), (-\cos(\beta_k)\cdot x_0(k) + sen(\beta_k)\cdot y_0(k))\right]^T$

$\boldsymbol{\mu}_{\beta,k} = \left[-sen(\beta_k), -kTsen(\beta_k), -\cos(\beta_k), -kT\cos(\beta_k), (sen(\beta_k)\cdot x_0(k) + \cos(\beta_k)\cdot y_0(k))\right]^T$

$\mathbf{f}_{\beta,k} = \mathbf{F}_{\beta,k-1} \cdot \mathbf{r}_{\beta,k}$

$\mathbf{F}_{\beta,k} = \mathbf{F}_{\beta,k-1} - (\mathbf{f}_{\beta,k} \cdot \mathbf{f}^T_{\beta,k} / (1 + \mathbf{r}^T_{\beta,k} \cdot \mathbf{f}_{\beta,k}))$

$\mathbf{W}_{\beta,k} = \mathbf{W}_{\beta,k-1} + \boldsymbol{\mu}_{\beta,k} \cdot \boldsymbol{\mu}^T_{\beta,k}$

$\boldsymbol{\varphi}^{(1)} = \boldsymbol{\theta}_{\beta,k-1}$

$\mathbf{c}^{(1)} = \mathbf{W}_{\beta,k} \cdot \boldsymbol{\varphi}^{(1)}$

$FN^{(1)}$ =Fator_de_Normalização= $\sqrt{\boldsymbol{\varphi}^{(1)T} \cdot \mathbf{c}^{(1)}}$

$\boldsymbol{\varphi}^{(0)} = -\infty, \forall j = 1$ -> $\boldsymbol{\varphi}^{(0)} = \left[-\infty, -\infty, -\infty, -\infty, -\infty,\right]$

while (($\left\|\boldsymbol{\varphi}^{(j)} - \boldsymbol{\varphi}^{(j-1)}\right\| > \eta$)e ($j < 1000$)) onde η é um valor pequeno e positivo

$\boldsymbol{\varphi}^{(j+1)} = \mathbf{F}_{\beta,k} \cdot \mathbf{c}^{(j)} / FN^{(j)}$

$\mathbf{c}^{(j+1)} = \mathbf{W}_{\beta,k} \cdot \boldsymbol{\varphi}^{(j+1)}$

$$FN^{(j+1)} = \sqrt{\boldsymbol{\varphi}^{(j+1)T} \cdot \mathbf{c}^{(j+1)}}$$

$$\boldsymbol{\varphi}^{(j+1)} = \boldsymbol{\varphi}^{(j+1)} \big/ FN^{(j+1)}$$

$$j = j + 1$$

end

$$\boldsymbol{\theta}_{\beta,k} = \boldsymbol{\varphi}^{(j)}$$

$$\boldsymbol{\mu}_{\beta,k} = \boldsymbol{\theta}_{\beta,k}(1:4) \big/ \theta_{\beta,k}(5)$$

end

Tabela 4.1: Extrato resumido do Algoritmo Estimador PL Original para um problema de BO-TMA.

4.3. Resumo do Algoritmo PL Modificado Implementado para o Método Estimador Pseudolinear Modificado

Como já foi dito anteriormente, o Método Estimador Pseudolinear Modificado utiliza o conceito de guardar o resultado com melhor aptidão e somente alterar o resultado obtido na iteração corrente caso este possua aptidão maior do que a encontrada até então.

A idéia de realizar esta modificação foi motivada após a verificação através de simulações de que o método original proposto [5] apresentava problemas de convergência, pois divergia e ficava preso eternamente no *loop* mais interno, principalmente quando o número de amostras é pequeno. Quando também são realizados testes com outras dinâmicas de movimento do contato e do navio observador, diferentes da única dinâmica de teste que foi apresentada no trabalho, os resultados não são satisfatórios.

Usando o conceito de guardar a aptidão, ou melhor, armazenar o resultado que obteve a melhor estimativa de posição do contato, encontrado até a posição anterior a do processamento atual, foram alcançados resultados melhores dos que os obtidos com o algoritmo original. Estes resultados serão apresentados no capítulo 6.

Um outro problema verificado foi a inexistência de um critério de saída ao se alcançar um número máximo de iterações, ou por tempo de processamento - *time-out*,

do *loop* mais interno, quando o algoritmo não apresenta a convergência esperada que possibilite encontrar uma solução satisfatória.

Deste modo, ocorria uma espera infinita pelo término do processamento dos dados, o que é inadmissível para o caso de processamento em tempo real para um sistema de acompanhamento de alvos, onde se deseja estimar a posição futura do mesmo com precisão e em pouco tempo.

A tabela 4.2 apresenta uma representação do fluxograma do PL Modificado. As linhas e itens que estão destacados em fundo amarelo representam as modificações que foram inseridas no algoritmo original, para que seja armazenado o melhor resultado.

Algoritmo estimador Modificado para o problema BO-TMA

1. Dados necessários para a execução do algoritmo estimador PL Modificado BO-TMA:

$(x_0(i), y_0(i))$, i = 0, 1, 2, ..., Posições conhecidas do navio observador.

β_i, i = 0, 1, 2, ..., Marcações conhecidas do contato.

2. Inicialização do algoritmo utilizando k_0 + 1 medidas.

$$\mathbf{r}_{\beta,i} = \left[\cos(\beta_i), iT\cos(\beta_i), -sen(\beta_i), -iTsen(\beta_i), (-\cos(\beta_i)\cdot x_0(i) + sen(\beta_i)\cdot y_0(i))\right]^T, \text{ i = 0, 1, 2, ..., } k_0$$

$$\boldsymbol{\mu}_{\beta,i} = \left[-sen(\beta_i), -iTsen(\beta_i), -\cos(\beta_i), -iT\cos(\beta_i), (sen(\beta_i)\cdot x_0(i) + \cos(\beta_i)\cdot y_0(i))\right]^T, \text{ i = 0, 1, 2, ..., } k_0$$

$$\mathbf{A}_{\mu,\beta} = \left[\mathbf{r}_{\beta,0}, \mathbf{r}_{\beta,1}, \mathbf{r}_{\beta,2}, \mathbf{r}_{\beta,3}, ..., \mathbf{r}_{\beta,k_0}\right]^T$$

$$\mathbf{R}_{\beta,k_0} = \mathbf{A}_{\mu,\beta}^T \cdot \mathbf{A}_{\mu,\beta}$$

$$\mathbf{W}_{\beta,k_0} = \sum_{i=0}^{k_0} \boldsymbol{\mu}_{\beta,i} \cdot \boldsymbol{\mu}_{\beta,i}^T$$

$\boldsymbol{\theta}_{\beta,i}$ = Autovetor Generalizado correspondente ao menor Autovalor obtido de eig($\mathbf{R}_{\beta,k_0}$, $\mathbf{W}_{\beta,k_0}$)

$$\mathbf{F}_{\beta,k_0} = \mathbf{R}_{\beta,k_0}^{-1}$$

3. *Loop* .principal onde se obtém as Estimativas

for k = k0 +1 , k0 +2 , ...

$$\mathbf{r}_{\beta,k} = \left[\cos(\beta_k), kT\cos(\beta_k), -sen(\beta_k), -kTsen(\beta_k), (-\cos(\beta_k)\cdot x_0(k) + sen(\beta_k)\cdot y_0(k))\right]^T$$

$$\boldsymbol{\mu}_{\beta,k} = \left[-sen(\beta_k), -kTsen(\beta_k), -\cos(\beta_k), -kT\cos(\beta_k), (sen(\beta_k)\cdot x_0(k) + \cos(\beta_k)\cdot y_0(k))\right]^T$$

$$\mathbf{f}_{\beta,k} = \mathbf{F}_{\beta,k-1}\cdot\mathbf{r}_{\beta,k}$$

$$\mathbf{F}_{\beta,k} = \mathbf{F}_{\beta,k-1} - (\mathbf{f}_{\beta,k}\cdot\mathbf{f}_{\beta,k}^T / (1+\mathbf{r}_{\beta,k}^T\cdot\mathbf{f}_{\beta,k}))$$

$$\mathbf{W}_{\beta,k} = \mathbf{W}_{\beta,k-1} + \boldsymbol{\mu}_{\beta,k}\cdot\boldsymbol{\mu}_{\beta,k}^T$$

$$\boldsymbol{\varphi}^{(1)} = \boldsymbol{\theta}_{\beta,k-1}$$

$$\mathbf{c}^{(1)} = \mathbf{W}_{\beta,k}\cdot\boldsymbol{\varphi}^{(1)}$$

$FN^{(1)}$=Fator_de_Normalização=$\sqrt{\boldsymbol{\varphi}^{(1)T}\cdot\mathbf{c}^{(1)}}$

$$\boldsymbol{\varphi}^{(0)} = -\infty, \forall j = 1 \text{ -> } \boldsymbol{\varphi}^{(0)} = \left[-\infty, -\infty, -\infty, -\infty, -\infty,\right]$$

while ($(\left\|\boldsymbol{\varphi}^{(j)} - \boldsymbol{\varphi}^{(j-1)}\right\| > \eta)$ e $(j < 1000)$)onde η é um valor pequeno e positivo

$$\boldsymbol{\varphi}^{(j+1)} = \mathbf{F}_{\beta,k}\cdot\mathbf{c}^{(j)} / FN^{(j)}$$

$$\mathbf{c}^{(j+1)} = \mathbf{W}_{\beta,k}\cdot\boldsymbol{\varphi}^{(j+1)}$$

$$FN^{(j+1)} = \sqrt{\boldsymbol{\varphi}^{(j+1)T}\cdot\mathbf{c}^{(j+1)}}$$

$$\boldsymbol{\varphi}^{(j+1)} = \boldsymbol{\varphi}^{(j+1)} / FN^{(j+1)}$$

$$j = j+1$$

end

if $(\left\|\boldsymbol{\varphi}^{(j)} - \boldsymbol{\varphi}^{(j-1)}\right\| < \eta)$ e $(j < 1000)$

$\boldsymbol{\theta}_{\beta,k}anterior = \boldsymbol{\varphi}^{(j)}$ - Guarda para futuro uso $\boldsymbol{\theta}_{\beta,k}$

$$\boldsymbol{\theta}_{\beta,k} = \boldsymbol{\varphi}^{(j)}$$

$$\boldsymbol{\mu}_{\beta,k} = \boldsymbol{\theta}_{\beta,k}(1:4) / \theta_{\beta,k}(5)$$

calcula as posições estimadas do contato

else

$\boldsymbol{\theta}_{\beta,k} = \boldsymbol{\theta}_{\beta,k}anterior$ - Recupera $\boldsymbol{\theta}_{\beta,k}anterior$ pois a solução não convergiu

$\boldsymbol{\mu}_{\beta,k} = \boldsymbol{\theta}_{\beta,k}(1:4)/\theta_{\beta,k}(5)$

calcula as posições estimadas do contato

end

end

Tabela 4.2: Resumo do Algoritmo Implementado para o Método Estimador Pseudolinear Modificado.

4.4. Emprego do Algoritmo de Estimação EKELUND para comparação com os algoritmos PL Original e PL Modificado

Baseado na dissertação [1] já citada anteriormente foi desenvolvido um script do MATLAB, utilizando o algoritmo de EKELUND proposto, que foi testado inicialmente como o mesmo conjunto de onze corridas simuladas utilizadas naquele trabalho.

Os dados de velocidade e de rumo verdadeiro do navio-patrulha perseguidor, e as amostras temporais das marcações verdadeiras para o contato, para duas janelas de tempo de aquisição em rumos e velocidades distintos, formando portanto duas pernadas distintas, são usados no calculo do alcance estimado do contato, dentro de uma determinada faixa de incerteza, para uma marcação verdadeira que corresponde a última marcação obtida para o contato.

Como foi visto anteriormente, neste método a trajetória de um contato, movendo-se supostamente em velocidade constante, pode ser aproximada por arcos de circunferências, desde que as linhas de base geradas pelo movimento do navio-patrulha perseguidor sejam muito menores do que a faixa de alcance de trabalho.

Assim, a distancia percorrida pelo navio observador deve ser muito menor do que a distância do navio observador ao contato durante o intervalo de medição. Portanto, o algoritmo de EKELUND fornece resultados satisfatórios quando esta condição é satisfeita, podendo apresentar resultados distorcidos quando o contato estiver muito próximo do navio observador.

As equações 3.1 e 3.2 permitem determinar os valores de $D_{\min}$ e D_{EK}, quando aplicadas aos dados a seguir, relativos a primeira pernada, e a primeira e segunda pernadas, respectivamente:

- Componente de velocidade transversal do movimento do contato com relação ao navio observador; e
- Série temporal de marcações, que na realidade é a taxa de variação da marcação com o tempo, ou seja, a primeira derivada da marcação com relação ao tempo que é portanto uma velocidade das marcações.

Estes valores determinam em conjunto com a marcação de referência ao final da segunda pernada o segmento de reta ($D_{EK} - D_{min}$) em que o contato esta provavelmente localizado, e que foi visto na figura 3.2.

4.5. Implementação da redução de ruído (*denoising*) com Wavelets

4.5.1. Função *wden* (*wavelet denoising*) da *toolbox* de *wavelets* do *MATLAB*

A redução do ruído (*denoising*) foi implementada através do uso da função *wden* (*wavelet denoising*) que pertence ao pacote de funções de *wavelets* do *MATLAB*.

A mesma fornece como saída um vetor XD com os dados filtrados após o *denoising* e possui as seguintes variáveis de entrada:

X – vetor de entrada com os dados que apresentam algum tipo de ruído adicionado;

TPTR – pode-se selecionar uma entre as quatro regras de *threshold*, que seguem o que foi dito no item 2.4.2:

- 'sqtwolog' - $\lambda = \sqrt{2 \cdot \sigma^2 \cdot \log n}$, portanto a equação 2.1;
- 'rigrsure' - *Stein's Unbiased Risk*, ou seja, é a equação 2.2;
- 'heursure' - Variante heurística das duas anteriores, ou seja, é feita uma composição das duas primeiras regras; e
- 'minimaxi' – Princípio de estatística para o projeto de estimadores, usando o valor mínimo do máximo erro médio quadrático obtido para a pior função de um determinado conjunto.

SORH – pode assumir os valores 's' ou 'h' que são associados ao *soft thresholding* ou ao *hard thresholding*. Considerando como exemplo para os dois casos um conjunto de pontos que representam a função y = x, teremos para o caso do *soft thresholding* um conjunto de pontos com apenas dois pontos de descontinuidade suave, enquanto que no caso do *hard thresholding* o resultado seria o de um conjunto de pontos com pelo menos quatro pontos de descontinuidade, sendo dois desses com forma abrupta.

SCAL – define um fator multiplicativo de re-escalamento do *threshold*:

- 'one' – para não utilizar este fator;
- 'sln' – para efetuar um re-escalamento usando uma estimativa do ruído baseada nos níveis iniciais dos coeficientes; e
- 'mln' - Estimativa dependente do nível de ruído – não branco;

N (*level)* – corresponde ao nível que se deseja chegar na decomposição de um sinal pelo processo de análise de *wavelet*, conforme foi visto na figura 2.3.

'wname' – define qual das famílias de *wavelets*, algumas já vistas no item 2.4.2, podem ser usadas, dentre as disponíveis no *MATLAB*:

- *Daubechies* – 'db1' ou *haar* até a 'db45';
- *Coiflets* – 'coif1'a 'coif5';
- *Symlets* ou *Symmlets* – 'sym2' a 'sym45';
- *Discrete Meyer* – 'dmey';
- *Biortogonal* – 'bior1.1', 'bior1.3', 'bior1.5','bior2.2', 'bior2.4', 'bior2.6', 'bior2.8' 'bior3.1', 'bior3.3', 'bior3.5', 'bior3.7' 'bior3.9', 'bior4.4', 'bior5.5', 'bior6.8'; e
- *Reverse Biorthogonal* – 'rbior1.1', 'rbior1.3', 'rbior1.5','rbior2.2','rbior2.4', 'rbior2.6', rbior2.8' 'rbior3.1', 'rbior3.3', 'rbior3.5', 'rbior3.7' 'rbior3.9', 'rbior4.4', 'rbior5.5', 'rbior6.8'.

4.5.2. Considerações para este trabalho no uso da função *wden*

Para as simulações foram consideradas as seguintes situações a fim de delimitar o espaço de busca:

Os dados de Latitude (Y) e Longitude (X) são tratados de modo separado. Assim sendo, calcula-se a raiz quadrada do erro médio quadrático (RMSE) dos pontos de

Latitude e Longitude com relação aos valores sem ruído. É efetuada a comparação dos resultados aplicando-se ou não a técnica de *wavelet denoising;*

Os níveis utilizados foram do dois ao nove;

A família de *wavelets* utilizada foi a *Symlets* (sym8), pois por ser simétrica apresentou os melhores resultados nos teste iniciais de *denoising* com relação as demais famílias. Possui ainda características similares as *wavelets* da família *Daubechies*, que são normalmente as mais comumente utilizadas; e

Um ruído gaussiano de 1% foi aleatoriamente somado aos dados simulados de Latitude e Longitude obtidos da simulação de um Movimento Retilíneo Uniforme (MRU) do contato. Este ruído foi gerado através do uso de um número pseudo-randômico gerado pela função *randn* do *MATLAB,* que possui distribuição normal de média zero e desvio padrão um. Depois de gerado o número pseudo-randômico, este foi dividido por cem e multiplicado pelo valor das coordenadas de Latitude e Longitude do contato. Ao final, adiciona-se o mesmo aos valores simulados respectivamente para a Latitude e para a Longitude do contato.

5. Padrões de Teste (*Benchmark*)

Foram utilizados para efeito de validação do algoritmo PL Modificado o mesmo conjunto de testes realizados na dissertação do IME [1] e também a mesma cinemática de teste utilizada no trabalho proposto para o PL Original que gerou o método BO-TMA [5].

Para validar o mesmo em outras situações, foram criadas outras cinemáticas de movimento dos navios, utilizando os mais variados rumos e velocidades, além das mais diversas condições iniciais.

Pelo fato de não existirem bases de dados com este tipo de informação nos repositórios de dados existentes na Internet, e pelo fato de que não existem referências ou bases de dados nos trabalhos publicados e pesquisados durante a revisão bibliográfica, optou-se por criarem-se estes vários tipos de cinemáticas de movimento de contatos e do navio observador, com o uso de duas pernadas, para comparar e validar o algoritmo PL Modificado, com relação ao algoritmo PL Original e ao algoritmo de EKELUND.

Pretende-se com isso, conseguir resultados melhores que os encontrados nos trabalhos estudados durante a revisão bibliográfica, bem como melhor aproximar a

teoria desenvolvida nesta dissertação de casos reais existentes nos cenários táticos operativos.

Os padrões de teste iniciais ajudaram a descobrir deficiências nos métodos estudados e também permitiram criar um método PL Modificado que teve sua eficiência comprovada através dos resultados obtido no próximo capítulo desta dissertação.

Aos conjuntos de dados utilizados nas simulações foram dadas as designações de cenários “A”, “B”, “C”, etc. Os cenários utilizados e suas condições iniciais estão resumidos nas tabela 5.1 a 5.5 que se encontram nas páginas seguintes:

Cenário	*Cinemáticas*	*Número de Conjunto de Dados*	*Resumo da Cinemática:*
"A"	*Trabalho BO-TMA [5]*	*1*	*Posição inicial do navio observador na origem (0,0;. Navio observador em movimento de zig-zag realizando curvas de 90 graus com velocidade de 25,4 nós; Rumo inicial 030 e posição inicial do contato na coordenada (7,7); e Velocidades baixas do contato.*
"B"	*Dissertação do IME [1]*	*5*	*Posição inicial do navio observador na origem (0,0); Navio observador em movimento único para todas a cinemáticas: Primeira pernada 040 graus com 9 nós e segunda pernada 280 graus com 12 nós; e Rumos, posição inicial e velocidades do contato conforme tabela 5.2.*
"C"	*Outras Cinemáticas de Movimento de Contato*	*5*	*Posição inicial do navio observador na origem (0,0); e Rumos, posição inicial e velocidades do contato conforme tabela 5.3.*
"D"	*Outras Cinemáticas de Movimento de Contato*	*5*	*Posição inicial do navio observador diferente da origem (0,0); e Rumos, posição inicial e velocidades do contato conforme tabela 5.4.*
"E"	*Outras Cinemáticas de Movimento de Contato*	*5*	*Posição inicial do navio observador na origem (0,0); e Rumos, posição inicial e velocidades do contato conforme tabela 5.5.*

Tabela 5.1: Relação dos cenários utilizados simulações.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0
B2	0	0	040	9,0	3,4	19,0	330	11,0	280	12,0
B3	0	0	040	9,0	5,0	27,0	330	11,0	280	12,0
B4	0	0	040	9,0	7,0	39,0	300	11,0	280	12,0
B5	0	0	040	9,0	0,1	17,0	330	4,0	280	12,0

Tabela 5.2: Resumo das condições iniciais para o cenário "B"

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
C1	0	0	000	13,0	7,0	7,0	030	9,0	090	9,0
C2	0	0	120	5,0	1,0	-9,0	248	16,0	200	17,0
C3	0	0	000	8,0	10,0	0,5	270	10,0	180	9,0
C4	0	0	100	11,0	20,0	1,0	170	12,0	330	9,0
C5	0	0	017	2,0	17,0	2,0	100	15,0	100	20,0

Tabela 5.3: Resumo das condições iniciais para o cenário "C"

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
D1	1	2	063	8,0	12,0	50,0	330	11,0	011	13,0
D2	25	47	078	5,0	63,0	15,0	045	12,0	045	12,0
D3	-12	45	265	7,0	65,0	-15,0	024	12,0	127	9,0
D4	1200	1500	254	10,0	1000,0	1510,0	056	14,0	301	9,0
D5	-5	-17	105	15,0	0,5	78,0	235	10,5	014	13,9

Tabela 5.4: Resumo das condições iniciais para o cenário "D"

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
E1	0	0	040	9,0	0,5	17,0	012	240,0	280	12,0
E2	0	0	040	9,0	2,0	12,0	325	120,0	280	12,0
E3	0	0	000	8,0	-10,0	15,0	270	110,0	330	15,0
E4	0	0	100	11,0	-20,0	-1,0	170	98,0	230	9,5
E5	0	0	317	4,5	17,0	-32,0	275	155,0	130	19,0

Tabela 5.5: Resumo das condições iniciais para o cenário "E"

6. Apresentação e Análise de Resultados

O presente trabalho é parte de um futuro sistema de acompanhamento de navios de superfície e de submarinos que se pretende implementar a bordo dos navios da MB. O mesmo foi inicialmente desenvolvido utilizando-se o programa MATLAB 7.1.0.246 (R14) - *Service Pack 3 for Windows XP*. Os scripts criados foram acoplados em uma única interface gráfica denominada: "Aplicativo BO-TMA". Os scripts passaram a formar módulos do presente aplicativo e são executados quando o usuário pressiona o botão correspondente à ação que deseja executar.

Os mesmos algoritmos foram também implementados usando as ferramentas de programação do pacote *MATLAB* 7.1.0.246 (R14) - *Service Pack 3 for Linux* rodando em um computador com sistema operacional *Linux*, empregando a distribuição *OpenSUSE 10.3*.

Inicialmente foi criado um módulo simulador de movimentos do contato com relação ao movimento do navio observador, com o objetivo de obter todas as informações possíveis sobre a cinemática em estudo, sem a necessidade de extração dos dados reais dos equipamentos existentes a bordo dos navios da MB.

Deste modo, informações como a posição do contato em qualquer instante de tempo, bem como a posição do navio observador após a mudança de rumo no início da segunda pernada, podem ser obtidas diretamente através das equações do MRU, para a Longitude e Latitude, ou melhor, para os eixos X e Y, respectivamente. No caso das marcações, que são os ângulos medidos a partir do NV até os sucessivos segmentos de reta imaginários que ligam o contato ao navio observador, estas são obtidas indiretamente através da medida do coeficiente angular de cada um destes segmentos de reta.

Além do módulo simulador de movimentos de alvos, também foram criados outros módulos que permitem interligar o presente aplicativo ao barramento de dados de um navio. As funções existentes são basicamente as de receber e guardar os dados em arquivos de dados, para posterior processamento, ou a de processamento *on-line* dos dados. Como estas funções fogem do escopo da presente dissertação, os mesmos não serão aqui tratados.

Na figura 6.1, é apresentada uma tela resultante de uma simulação. O "Aplicativo BO-TMA" apresentada originalmente uma *interface* colorida, que é a saída

padrão do *MATLAB*. Os gráficos podem *plotar* até oitos séries de dados em cores distintas, para melhor visualização do usuário. Existe também o recurso de associar a cada ponto pertencente a uma série determinada um símbolo gráfico como marcador, isto é, usando sinais de '+', '*' ou 'x', por exemplo.

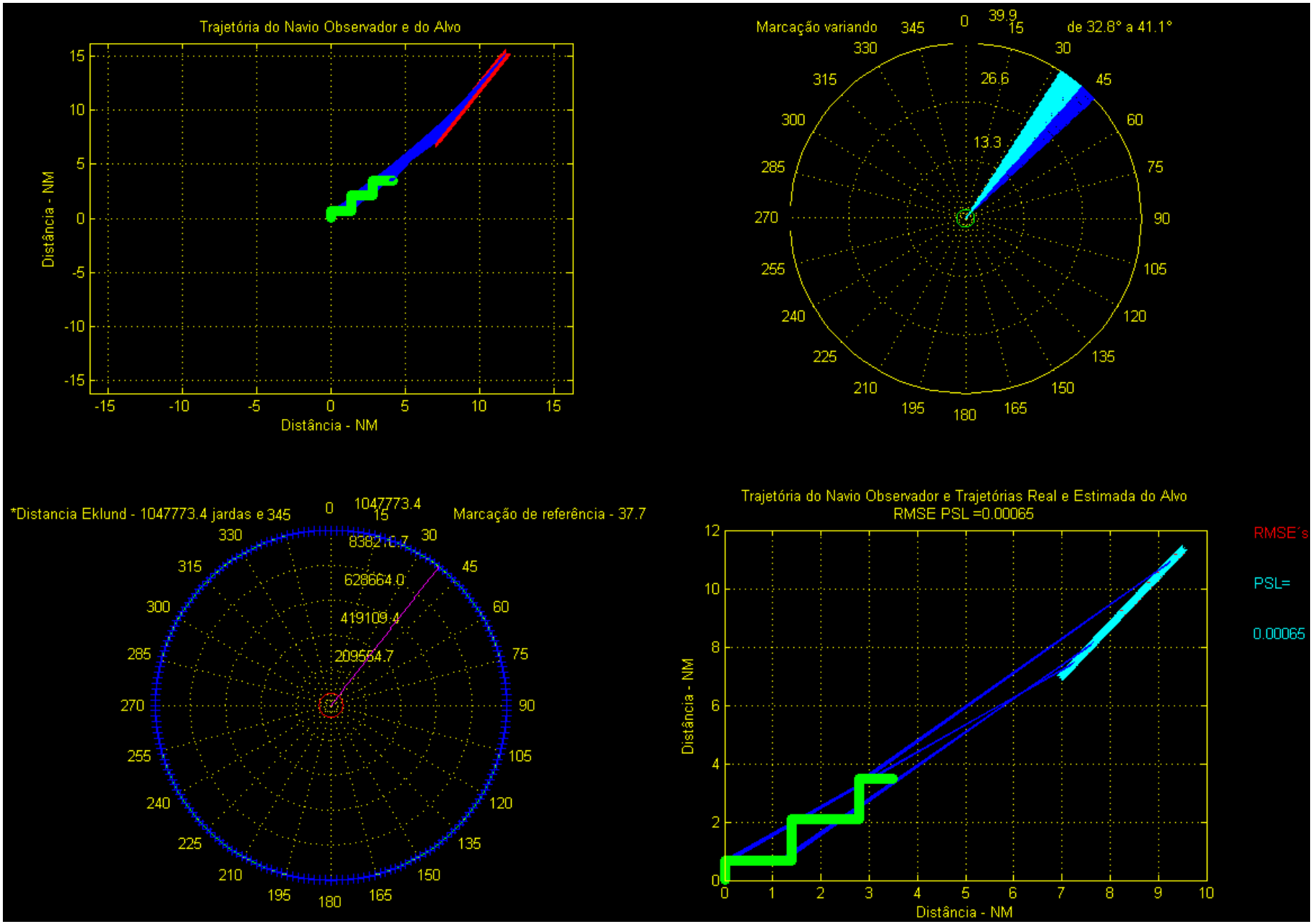


Figura 6.1: Tela de interface com o usuário do Sistema de Acompanhamento de Navios desenvolvido utilizando o pacote *MATLAB*

Pode-se verificar que nesta figura existem quatro grupos de gráficos que são:

1. No canto superior esquerdo, o gráfico resultante da cinemática de movimento do alvo e do navio observador, segundo as equações do MRU. Os pontos ocupados pelo contato no plano XY são representados pelo símbolo "+", originalmente na cor vermelha. Já as sucessivas coordenadas de Longitude e Latitude que o navio observador ocupou durante a simulação são representados pelo símbolo gráfico "O" na cor verde. As linhas azuis representam os segmentos de reta que ligam o navio observador ao contato, e que são as linhas de visada para o contato. Como já foi dito anteriormente, o coeficiente angular destes segmentos de reta são as marcações do contato que são obtidas pelo navio observador. As mesmas foram geradas de modo a permitir uma melhor visualização e entendimento do movimento do contato por parte do usuário, sendo que existe um botão de seleção no canto inferior direito que permite ao usuário optar ou não por este modo de visualização. Por fim, o mesmo dá uma idéia do cenário tático visualizado por um observador localizado acima do teatro de operações (TO), ou seja, como se fosse a visão de um observador sobrevoando a grande altitude o TO a bordo de um helicóptero ou avião de vigilância radar;
2. No canto superior direito, o gráfico resultante, que foi gerado em coordenadas polares, representa as marcações do contato obtidas a partir dos coeficientes angulares dos segmentos de reta que ligam o navio observador ao contato. Este tipo de gráfico melhor representa o tipo de saída gráfica existente na maioria dos equipamentos e sistema de sensores existentes nos navios, como por exemplo, os equipamentos de sonar, de radar e de MAGE. Os segmentos de reta na cor azul escura representam as marcações relativas a primeira pernada e os que estão na cor azul clara, ou ciano, representam as marcações da segunda pernada;
3. O gráfico existente no canto inferior esquerdo representa a estimativa resultante do algoritmo de EKELUND para o movimento do contato, após o final da segunda pernada. Nele podemos observar que é fornecido um segmento de reta ($D_{EK} - D_{min}$) na cor violeta, representando uma linha imaginária sobre a qual deve estar localizado o contato. Ainda com relação a este gráfico, que foi gerado em coordenadas polares, também são fornecidos numericamente o valor da distância EKELUND e a marcação final para o contato; e

4. Por fim, no canto inferior direito, está apresentado o resultado da estimativa gerado pelo algoritmo PL Original, para a simulação de movimento do contato e do navio observador que foi proposto no trabalho original [5]. A posição estimada do contato é marcada no mesmo pelo símbolo "*" na cor azul claro (ciano), sendo que as sucessivas posições verdadeiras do contato que foram simuladas com o uso das equações do MRU são representadas novamente pelo símbolo "+" na cor vermelha. As sucessivas coordenadas de Longitude e Latitude que o navio observador ocupou durante a simulação são representados pelo símbolo gráfico "O" na cor verde. Novamente os segmentos de retas ligando o contato ao navio observador e que acabam fornecendo as marcações do contato com relação ao observador podem ser visualizadas na cor azul escuro pelo usuário através da seleção do correspondente botão no canto inferior direito. No caso acima, durante o processamento dos dados, o usuário ligou e desligou a seleção do referido botão, e foram *plotadas* somente algumas linhas de marcação no referido gráfico.

Em todos os casos que foram simulados, a distância do contato ao navio observador é calculada através da equação abaixo:

$$Distância_{do_alvo_ao_observador} = \sqrt{\left(x_C(k) - x_o(k)\right)^2 + \left(y_C(k) - y_o(k)\right)^2} \tag{6.1}$$

6.1. Simulação do Cenário "A" utilizando o Algoritmo PL Original

A figura 6.2 representa o movimento original do contato e do navio observador conforme foi proposto em [5]. Os gráficos foram intencionalmente convertidos para o padrão preto e branco, conforme previsto nas normas em vigor.

Na figura 6.3, encontram-se as ampliações do resultado, para melhor entendimento, da simulação com a cinemática do trabalho BO-TMA utilizando o algoritmo PL Original. Encontram-se identificadas pelos respectivos símbolos de '+' e '*' e circundados por círculos, duas posições do contato, sendo uma real e a outra estimada.

A tabela 6.1 apresenta um resumo dos testes efetuados para o cenário "A" utilizando-se os algoritmos EKELUND e PL Original. Já na tabela 6.2, estão resumidas as condições inicias para o problema simulado.

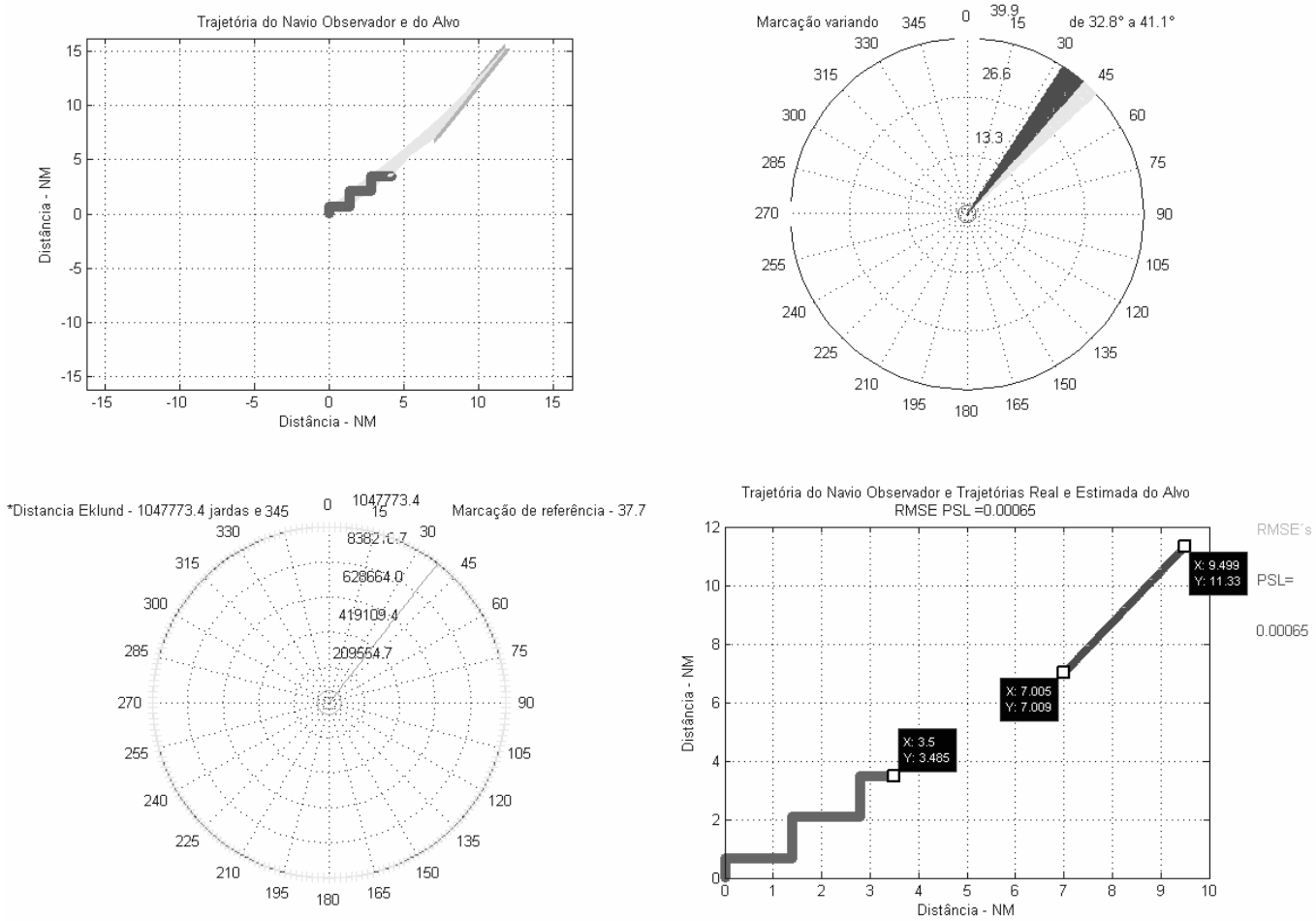


Figura 6.2: Resultado da Simulação com Cenário "A" utilizando o Algoritmo PL Original

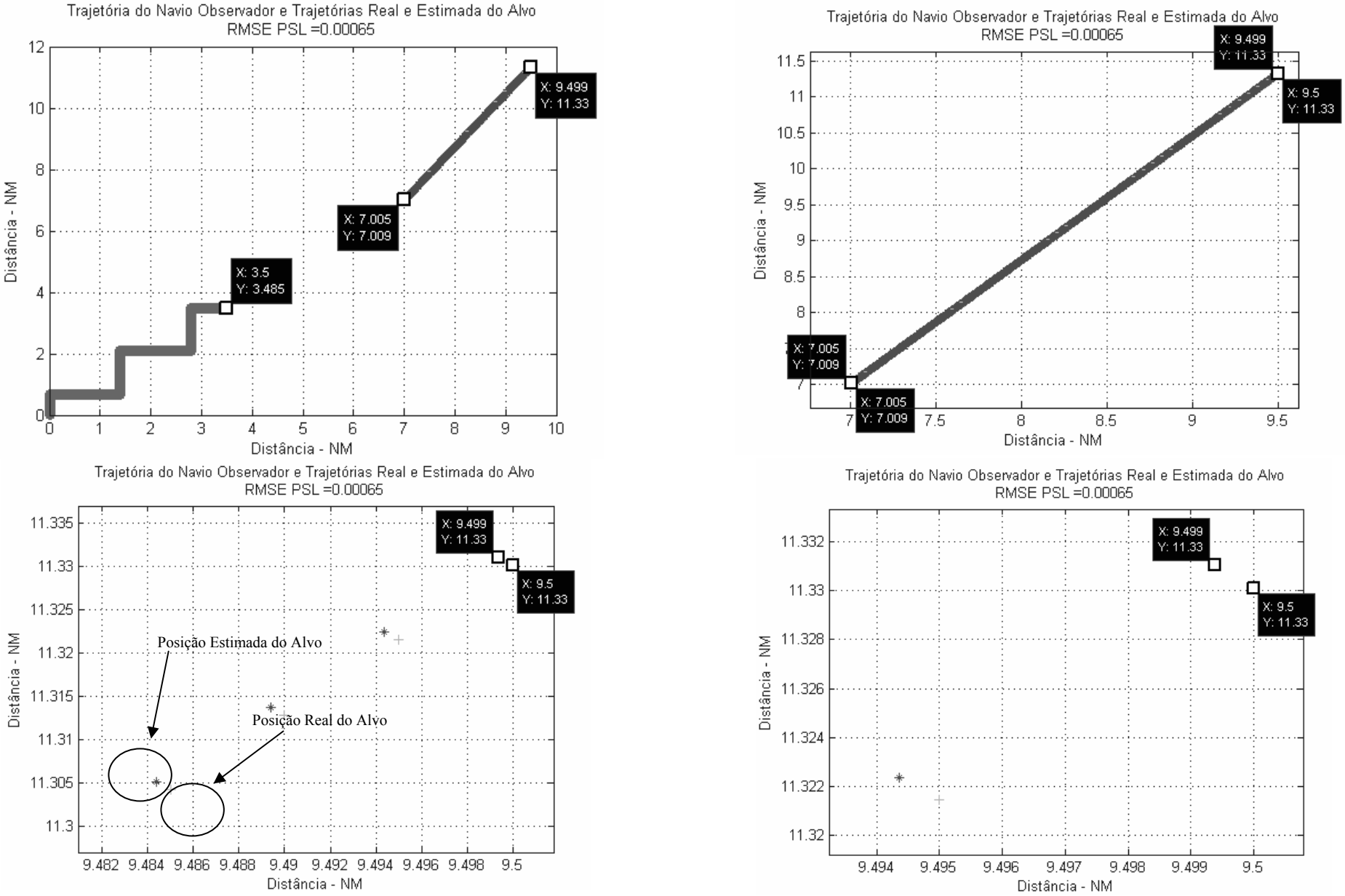


Figura 6.3: Ampliações do Resultado da Simulação com Cenário "A" utilizando o Algoritmo PL Original

D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*			
		X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	$D_{PL\text{-}MOD}$	$RMSE_{PL}$
jardas	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
1047773,4	517,3	$X_T = 9,5$ $X_O = 3,5$	$Y_T = 11,33$ $Y_O = 3,485$	9,876	9,499	11,330	9,876	0,00065

Tabela 6.1: Resumo dos testes efetuados para o Cenário "A" utilizando o Algoritmo PL Original.

Posição inicial do navio observador X,Y - (NM)	(0 , 0)
Rumo inicial do navio observador na primeira pernada (graus)	000
Instantes de tempo em que o navio observador muda seu rumo (guinada) de 90 graus	$t = 50 + 100 \cdot i$ $i = 0, 1, 2, 3, ...$
Velocidade do navio observador na primeira pernada (nós)	25,4
Posição inicial do contato X,Y - (NM) – para o simulador	(7 , 7)
Rumo do contato na primeira e segunda pernadas (graus)	030
Velocidade do contato na primeira e segunda pernada (nós)	18,0
Intervalo de tempo amostragem em segundos (s)	2
Tempo total de amostragem (Janela de tempo) em segundos (s)	1000
Rumo do navio observador na segunda pernada	090
Velocidade do navio observador na segunda pernada	5,0

Tabela 6.2: Resumo das Condições Iniciais para o Cenário "A" utilizando o Algoritmo PL Original.

Pode-se observar na tabela 6.1 que os resultados alcançados são bem próximos, com relação à distância final do contato, e compatíveis com o resultado alcançado em [5], que foi próximo de zero após cerca de 200 iterações, conforme fig 3 do mesmo.

Os resultados de EKELUND apresentam um erro grande, pois o rumo e velocidades na primeira pernada não foram constantes, devido ao *zig-zag* do navio observador.

6.2. Simulação dos Cenários “B" utilizando os Algoritmos EKELUND, PL Original e PL Modificado.

6.2.1. Simulação do Cenário “B1" utilizando o Algoritmo PL Original com intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente.

Na tabela 6.3, estão resumidas as condições inicias para o simulador MRU gerar o primeiro conjunto de dados utilizados na dissertação do IME [1], ou seja, o cenário “B1”. O valor exato da posição inicial do contato e de sua velocidade não foram fornecidas no trabalho citado, somente as séries temporais das marcações para a primeira pernada e para a segunda pernada. Deste modo, foram feitas diversas aproximações através de simulações de forma a conseguir reproduzir as mesmas séries temporais das marcações com um erro pequeno.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0

Tabela 6.3: Resumo das condições iniciais para o simulador MRU gerar o cenário “B1”.

Os gráficos ampliados com o resultado da simulação são mostrados na figura 6.4, para melhor entendimento. As simulações foram feitas com o cenário “B1” utilizando o algoritmo PL Original, considerando-se que o intervalo de amostragem e o tempo total de amostragem (janela de tempo) são iguais a 12s e 240s, respectivamente. Novamente as posições do contato, sendo uma real e a outra estimada, encontram-se identificadas pelos respectivos símbolos de ‘+’ e ‘*’.

Pode-se observar por estes gráficos que o erro de estimação foi elevado.

A tabela 6.4 apresenta um resumo dos testes efetuados para o primeiro conjunto de dados.

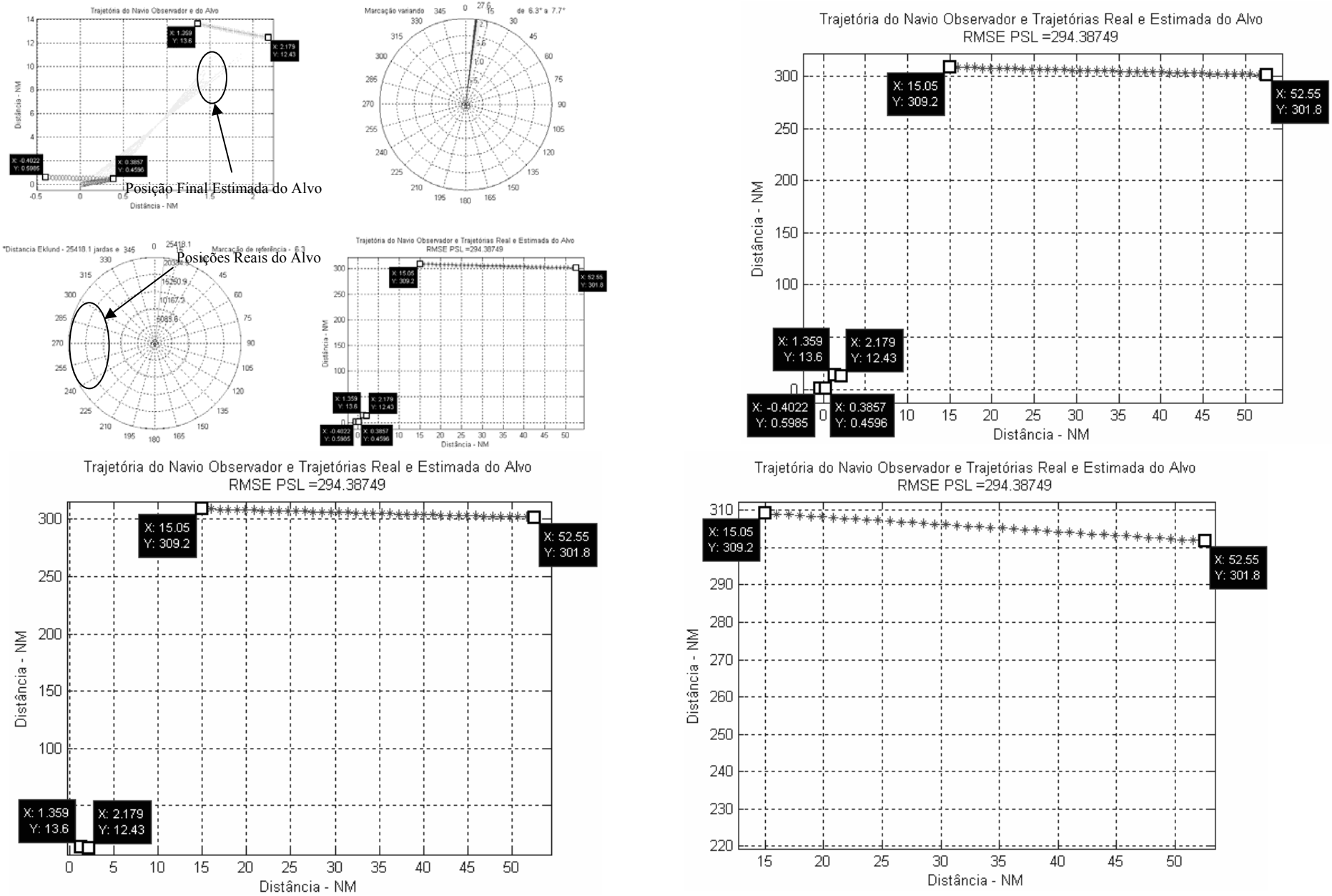


Figura 6.4: Resultado da Simulação para o cenário "B1" utilizando os Algoritmos EKELUND e PL Original.

Cenário	*Marcações iniciais e finais das pernadas*	*Dmin*		D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*			
						X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	$D_{PL\text{-}MOD}$	$RMSE_{PL}$
	Graus	***jardas***	**NM**	***jardas***	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
B1	1º pernada [10 a 6,3] 2º pernada [4 a 5,8]	10411,0	5,14	25310,0 Simul 25418,1	12,50 Simul 12,55	$X_T = 1,359$ $X_O = -0,402$	$Y_T = 13,60$ $Y_O = 0,599$	13,119	15,05	309,20	308,945	294,387

Tabela 6.4: Resumo dos testes efetuados para o cenário "B1" utilizando os Algoritmos EKELUND e PL Original.

Pode-se observar na tabela 6.4 que os resultados alcançados estão muitos distantes, com relação à distância final real do contato. A observação que o erro $RMSE_{PL}$ foi muito alto, e o fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real, mostra que o algoritmo PL Original não apresentou a convergência esperada. Por isso, foram executadas análises nos resultados internos do algoritmo.

Foi verificado que nas iterações iniciais existia uma convergência para um valor estimado correto, mas como não havia um mecanismo de memorização deste resultado, o algoritmo normalmente ficava preso em um mínimo local eternamente. Neste ponto foi acrescentado um critério de parada por número máximo de iterações, que quando alcançado, retornava uma solução com erro de estimativa elevado.

Por fim, foi verificado que, quando a estimativa correta da iteração anterior era guardada e novamente utilizada, no caso do algoritmo não convergir para a iteração atual, o resultado encontrado apresentava uma significativa diminuição no erro RMSE. A esta alteração no modo de funcionamento do algoritmo foi dado o nome de PL Modificado.

Para validar o mesmo, foram feitas outras simulações que são apresentadas ao longo deste capítulo.

6.2.2. Simulação do Cenário "B1" utilizando o Algoritmo PL Modificado com intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente.

Na tabela 6.5, estão resumidas as condições inicias para o simulador MRU gerar o cenário "B1".

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0

Tabela 6.5: Resumo das condições iniciais para o simulador MRU gerar o cenário "B1".

Os gráficos ampliados com o resultado da simulação são mostrados na figura 6.5, para melhor entendimento. As simulações foram feitas com o mesmo cenário "A", referente ao primeiro conjunto de dados utilizados na Dissertação do IME [1], utilizando agora o algoritmo PL Modificado.

Continua-se a considerar que o intervalo de amostragem e o tempo total de amostragem (janela de tempo) são iguais a 12s e 240s, respectivamente.

Novamente as posições do contato, sendo uma real e a outra estimada, encontram-se identificadas pelos respectivos símbolos de '+' e '*'.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo modificado foi muito menor do que o erro obtido para o algoritmo original.

A tabela 6.6 apresenta um resumo dos testes efetuados para o primeiro conjunto de dados.

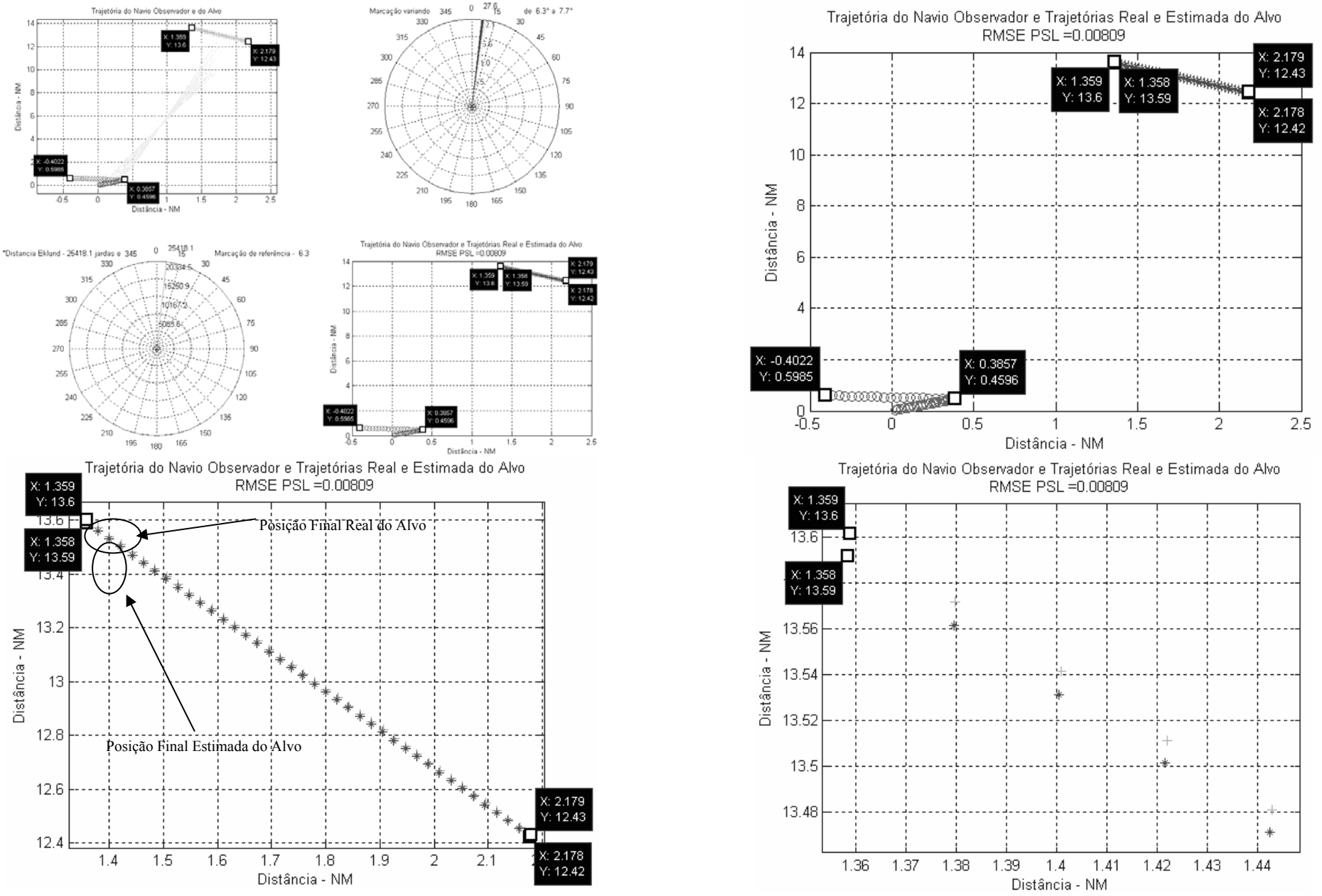


Figura 6.5: Resultado da Simulação para o cenário "B1" utilizando os Algoritmos EKELUND e PL Modificado

Cenário	*Marcações iniciais e finais das pernadas*	*Dmin*		D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
						X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	***jardas***	**NM**	***jardas***	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
B1	1º pernada [10 a 6,3] 2º pernada [4 a 5,8]	10411,0	5,14	25310,0 Simul 25418,1	12,50 Simul 12,55	$X_T = 1,359$ $X_O = -0,402$	$Y_T = 13,60$ $Y_O = 0,599$	13,119	1,358	13,591	13,112	0,0081

Tabela 6.6: Resumo dos testes efetuados para o cenário "B1" utilizando os Algoritmos EKELUND e PL Modificado.

Pode-se observar na tabela 6.6 que os resultados alcançados estão muito próximos, com relação à distância final real do contato. A observação que o erro $RMSE_{PL-MOD}$ foi praticamente igual a zero, e o fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real e da estimativa alcançada pelo algoritmo PL Modificado, mostra que o mesmo apresentou uma convergência muito melhor do que a esperada.

Para validar o mesmo em outros casos, foram feitas outras simulações, que são apresentadas a seguir, com os conjuntos de dados restantes da dissertação do IME [1], ou seja, utilizando os cenários "B1" a "B5".

6.2.3. Simulação dos Cenários “B1" a “B5" - intervalo de tempo e tempo total de amostragem (janela de tempo) de 12s e 240s, respectivamente.

Estão resumidas na tabela 6.7 as condições inicias que são inseridas no simulador MRU a fim de sejam gerados os cenários “B1” a “B5”. Foram feitas diversas aproximações através de simulações de forma a conseguir reproduzir as mesmas séries temporais das marcações com um erro aceitável.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0
B2	0	0	040	9,0	3,4	19,0	330	11,0	280	12,0
B3	0	0	040	9,0	5,0	27,0	330	11,0	280	12,0
B4	0	0	040	9,0	7,0	39,0	300	11,0	280	12,0
B5	0	0	040	9,0	0,1	17,0	330	4,0	280	12,0

Tabela 6.7: Resumo das condições iniciais para o simulador MRU gerar os cenários “B1” a “B5” - Algoritmos EKELUND, PL Original e PL Modificado.

Os gráficos ampliados do quarto conjunto de dados com os resultados das simulações são mostrados nas figuras 6.6 e 6.7, para melhor entendimento. As simulações foram feitas utilizando tanto o algoritmo PL Original, quanto o algoritmo PL Modificado, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem e o tempo total de amostragem (janela de tempo) continuam iguais a 12s e 240s, respectivamente, nas simulações realizadas.

Novamente as posições do contato, sendo uma real e a outra estimada, encontram-se identificadas pelos respectivos símbolos de ‘+’ e ‘*’.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo modificado foi muito menor do que o erro obtido para o algoritmo original. A tabela 6.8 apresenta um resumo dos testes efetuados para o os conjuntos de dados.

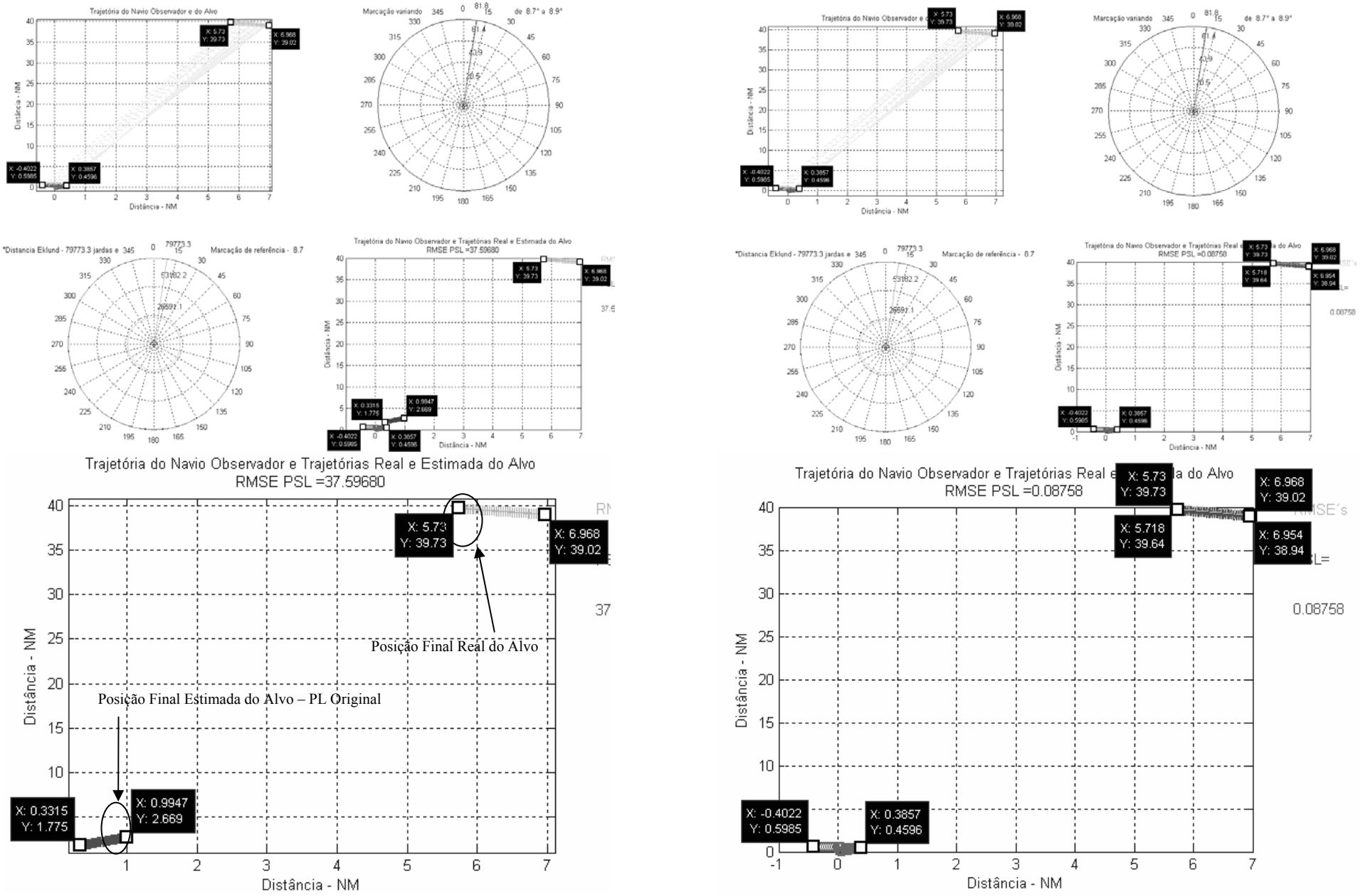
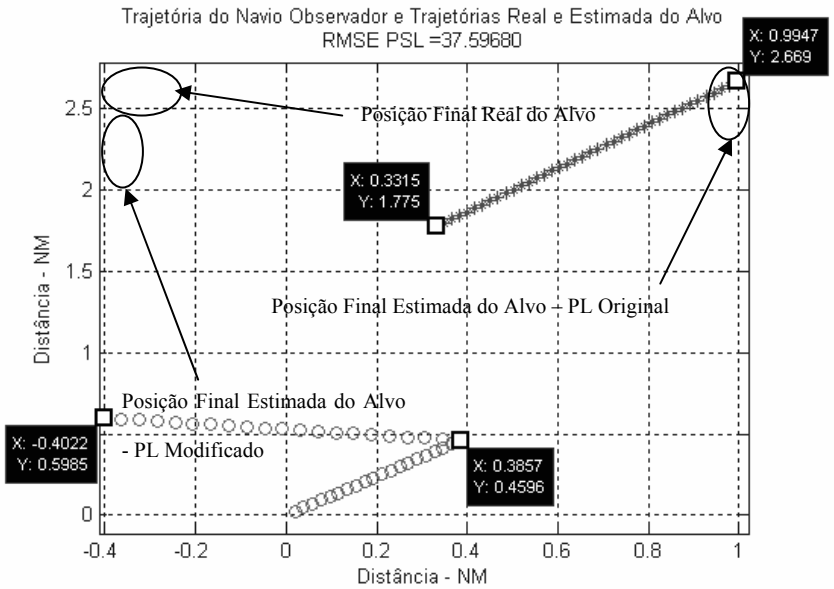
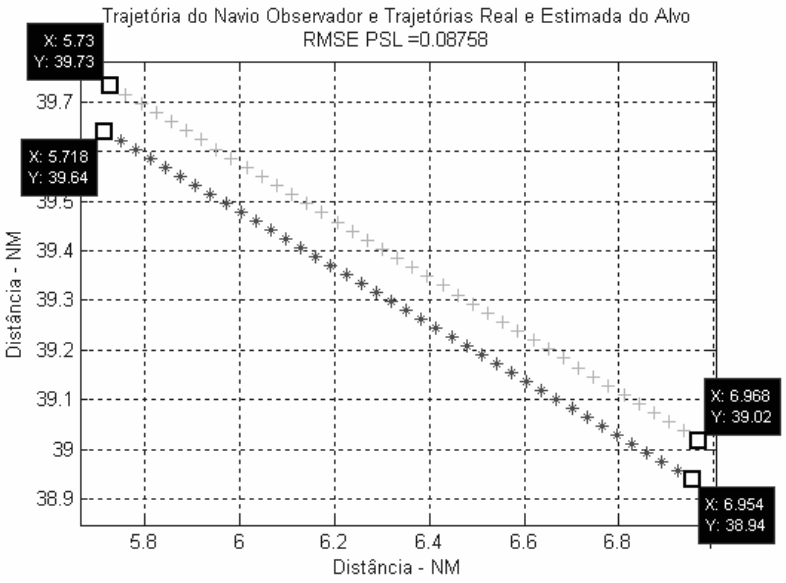
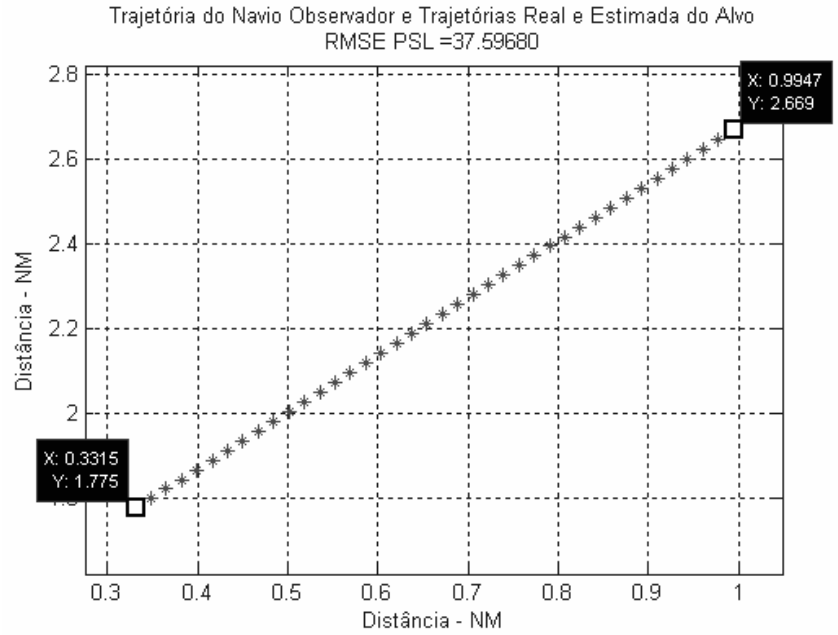


Figura 6.6: Resultado da Simulação para o cenário "B4" utilizando os Algoritmos EKELUND, PL Original e PL Modificado







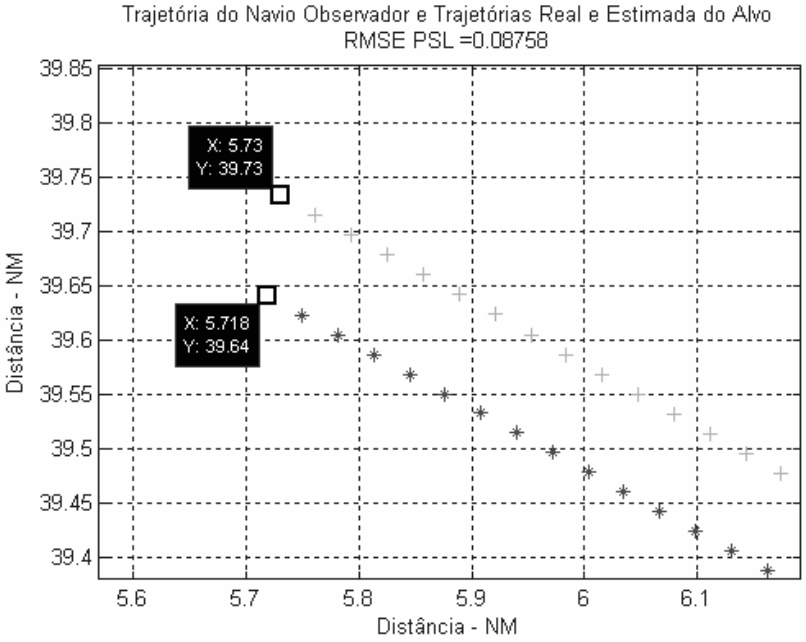


Figura 6.7: Resultado da Simulação para o cenário "B4" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

Pode-se observar na tabela 6.8 que os resultados alcançados pelo algoritmo estimador PL Modificado em termos da distância final estimada do contato, para todos os conjuntos de dados, estão muito próximos da distância final real do contato.

No entanto, o algoritmo PL Original apresentou erros $RMSE_{PL}$ muito grandes, não só com relação às distâncias estimadas, mas também com relação aos rumos estimados e as velocidades estimadas do contato, uma vez que, apesar da trajetória estimada pertencer à família de possíveis soluções do problema, o contato estaria navegando com rumo invertido de quase 180º e velocidade menor do que a real.

Entretanto, se for considerado que todas as linhas de visada, que são as retas suporte a partir das quais se medem as marcações, passam pela posição estimada do contato, chega-se a conclusão que o erro de distância provocaria um erro no cálculo da distância, na solução do tiro, mas não um erro em termos da marcação final estimada. Este fato não seria conveniente, uma vez que o armamento, mesmo que possuindo processamento e navegação embarcados, seria inicializado com condições inicias incorretas, o que provavelmente provocaria a perda do contato devido a um engajamento deficiente.

A observação que o erro $RMSE_{PL\text{-}MOD}$ foi praticamente igual a zero, e o fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real e também da estimativa alcançada pelo algoritmo PL Modificado, demonstra que o mesmo funcionou corretamente em todos os casos, apresentando uma convergência muito melhor do que a esperada.

Foram executadas outras simulações, mantendo-se os mesmos conjuntos de dados, mas alterando-se o intervalo de amostragem de 12 s para 2s e mantendo-se o tempo total de amostragem (janela de tempo) igual a 240s, pois foi verificado que em algumas situações em que as condições inicias do contato, em termos de posição inicial, rumo e velocidade, havia uma melhora na convergência para o algoritmo PL Original. Como a massa de dados utilizada no trabalho [5] apresenta um número de pontos igual a 500, para um intervalo de amostragem de 2s e um tempo total de amostragem igual a 1000s, conforme podemos observar na tabela 6.2 deste capítulo, foi vislumbrada a possibilidade de que o algoritmo PL Original, por ser inicializado com 10% dos dados, apresentaria uma sensibilidade maior quando o número de amostras utilizadas para a sua inicialização fosse pequeno.

Pretende-se com isso comprovar que o algoritmo PL Modificado possui uma maior robustez do que o PL Original, principalmente quando o número de amostras for menor, ou ainda se um erro no processo de leitura das marcações for propositalmente inserido, sob a forma de ruído.

Cenário	*Marcações iniciais e finais das pernadas*	*Dmin*		D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*				*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
						X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	D_{PL-MOD}	$RMSE_{PL}$	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	***jardas***	**NM**	***jardas***	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
B1	1º pernada [10,1 a 6,3] 2º pernada [6,3 a 7,7]	10411,0	5,14	25310,0	12,50	$X_T = 1,359$ $X_O = -0,402$	$Y_T = 13,60$ $Y_O = 0,599$	13,119	15,05	309,2	308,945	294,387	1,358	13,591	13,112	0,0081
				Simul 25418,1	Simul 12,55											
B2	1º pernada [10,1 a 7,9] 2º pernada [7,9 a 8,9]	20360,0	10,05	38294,0	18,91	$X_T = 2,667$ $X_O = -0,402$	$Y_T = 20,27$ $Y_O = 0,599$	19,909	0,696	0,149	1,187	20,136	2,608	19,850	19,485	0,3061
				Simul 39108,6	Simul 19,31											
B3	1º pernada [10,5 a 8,9] 2º pernada [8,9 a 9,6]	45813,0	22,61	54080,0	26,70	$X_T = 4,267$ $X_O = -0,402$	$Y_T = 28,27$ $Y_O = 0,599$	28,062	30,85	236,5	237,962	209,767	4,277	28,380	28,172	0,1456
				Simul 55613,6	Simul 27,46											
B4	1º pernada [10,2 a 8,7] 2º pernada [8,7 a 8,9]	45813,0	22,61	79739,0	39,37	$X_T = 5,73$ $X_O = -0,402$	$Y_T = 39,73$ $Y_O = 0,599$	39,609	0,9947	2,669	2,497	37,597	5,718	39,640	39,518	0,0876
				Simul 79733,3	Simul 39,37											
B5	1º pernada [0,3 a 358,6] 2º pernada [358,6 a 0,8]	26033,0	12,85	34513,0	17,04	$X_T = -0,1667$ $X_O = -0,402$	$Y_T = 17,46$ $Y_O = 0,599$	16,871	4,799	-70,18	70,970	88,892	0,167	17,460	16,871	0,0032
				Simul 34150,2	Simul 16,86											

Tabela 6.8: Resumo dos testes efetuados para os cenários "B" utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 12s e 240s, respectivamente.

6.2.4. Simulação dos Cenários “B1" a “B5" - intervalo de tempo e tempo total de amostragem (janela de tempo) de 2s e 240s, respectivamente.

Estão resumidas na tabela 6.9 as mesmas condições inicias do item anterior que são inseridas no simulador MRU a fim de sejam gerados os cenários “B1” a “B5”.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0
B2	0	0	040	9,0	3,4	19,0	330	11,0	280	12,0
B3	0	0	040	9,0	5,0	27,0	330	11,0	280	12,0
B4	0	0	040	9,0	7,0	39,0	300	11,0	280	12,0
B5	0	0	040	9,0	0,1	17,0	330	4,0	280	12,0

Tabela 6.9: Resumo das condições iniciais para o simulador MRU gerar os cenários “B1” a “B5” - Algoritmos EKELUND, PL Original e PL Modificado.

Os gráficos ampliados do quarto conjunto de dados com os resultados das novas simulações são mostrados nas figuras 6.8 e 6.9, para melhor entendimento. As simulações foram feitas utilizando tanto o algoritmo PL Original, quanto o algoritmo PL Modificado, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem foi alterado de 12 s para 2s, mas foi mantido o tempo total de amostragem (janela de tempo) igual a 240s, de forma a comprovar o algoritmo PL Original é mais sensível do que o PL Modificado quando são alteradas levemente as condições inicias do contato, em termos de posição inicial, rumo e velocidade, pois ocorria uma pequena melhora na convergência com relação ao algoritmo PL Original.

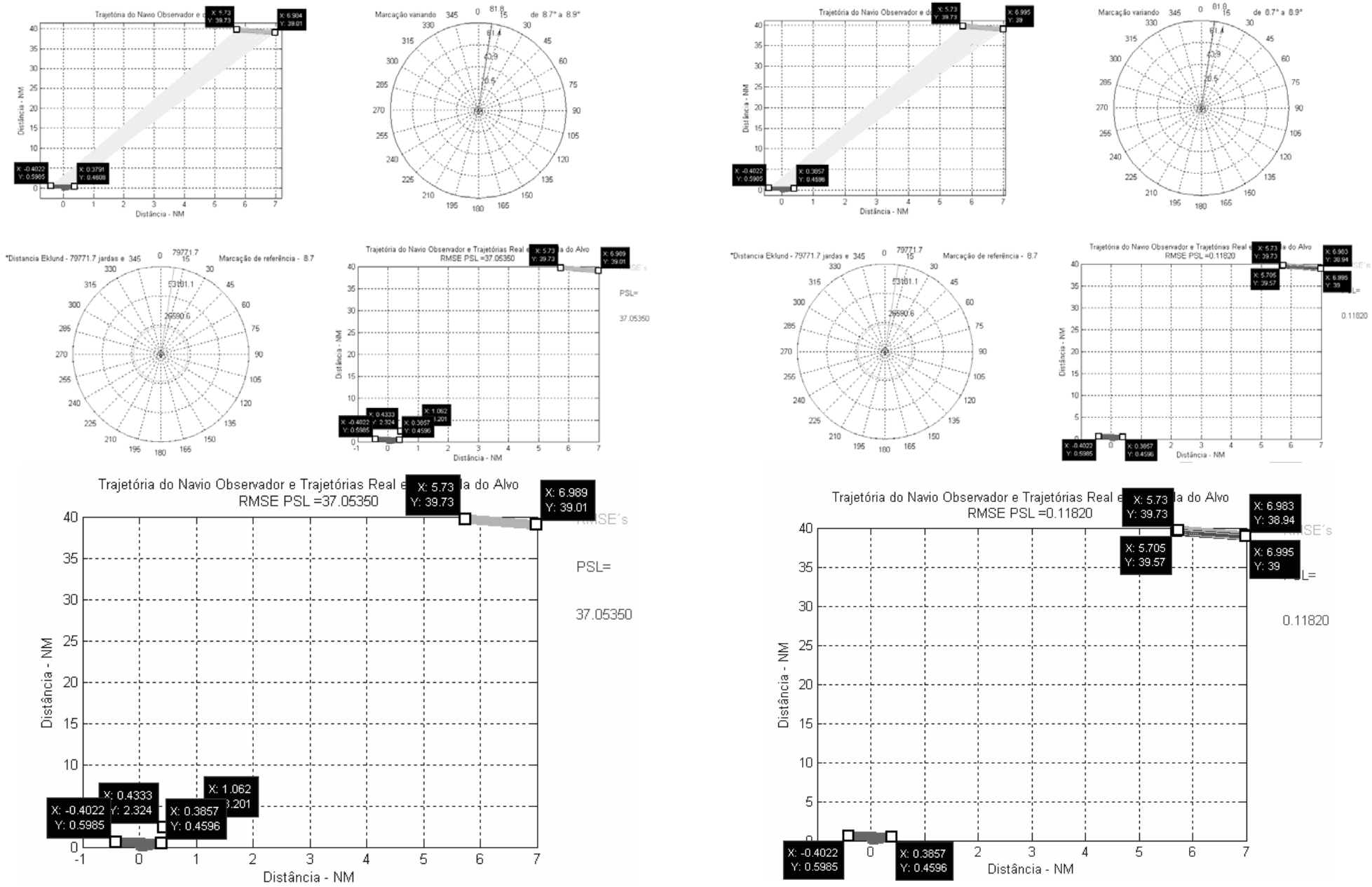
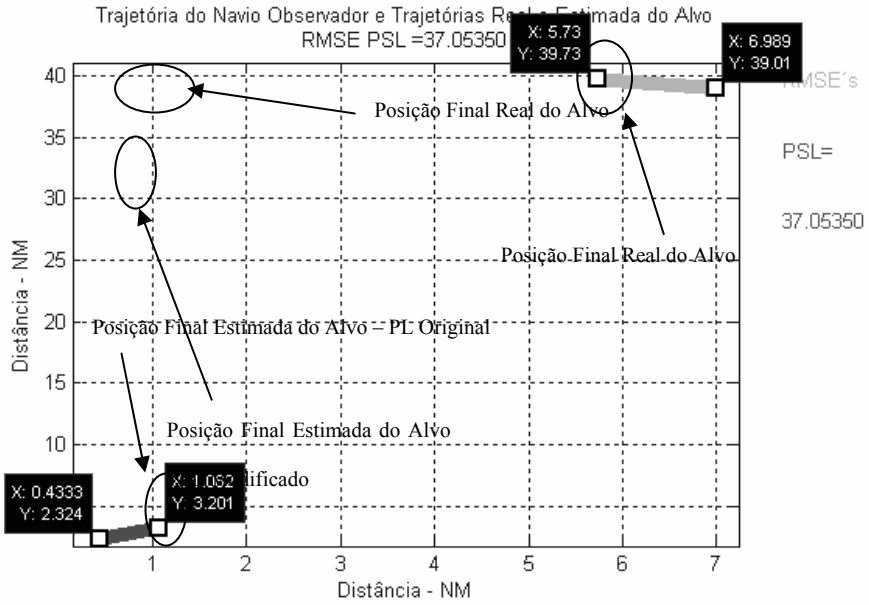
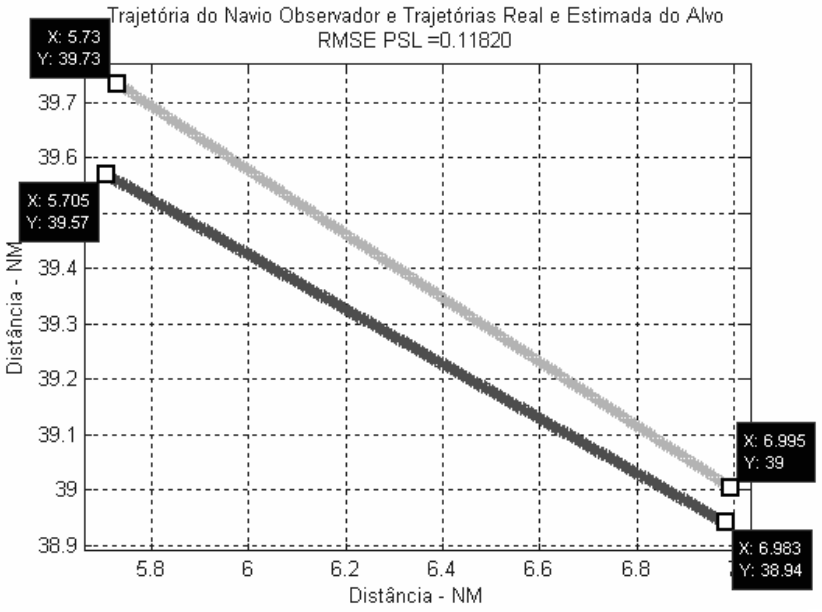
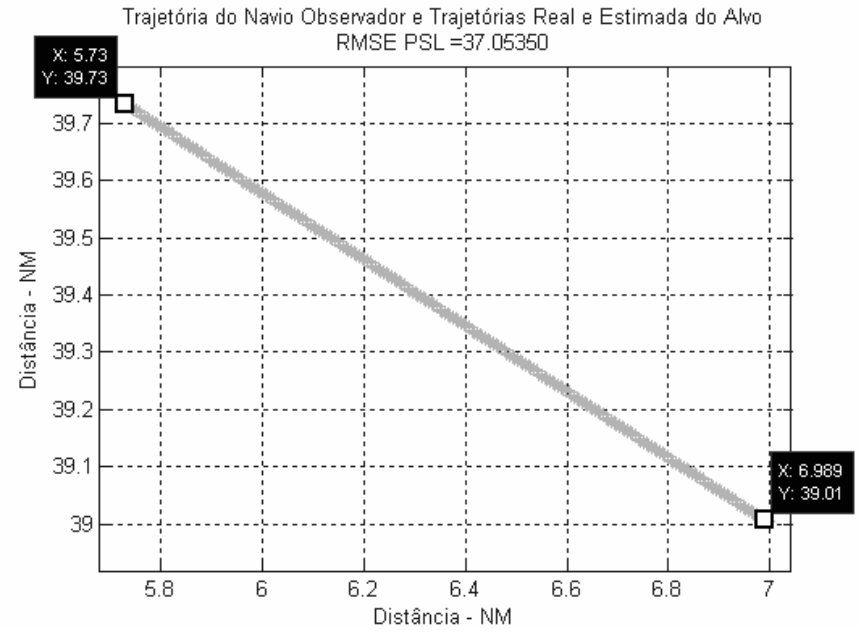


Figura 6.8: Resultado da Simulação para o cenário "B4" utilizando os Algoritmos EKELUND, PL Original e PL Modificado







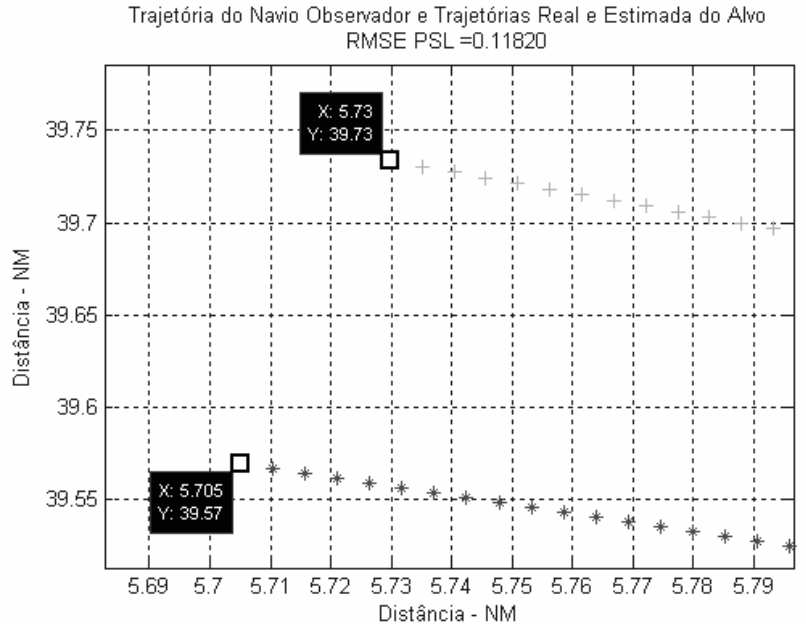


Figura 6.9: Resultado da Simulação para o cenário "B4" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo modificado continuou a ser muito menor do que o erro obtido para o algoritmo original.

A tabela 6.10 apresenta um resumo dos testes efetuados para os conjuntos de dados. Comparando a tabela 6.10 com a tabela 6.8, pode-se verificar que o erro $RMSE_{PL\text{-}MOD}$ diminuiu para todos os conjuntos de dados, a exceção do quarto conjunto, onde o erro aumentou um pouco, passando de 0,0876 para 0,1182.

Já o erro $RMSE_{PL}$, apresentou uma diminuição significativa em três conjuntos de dados, que são o um, o três e o cinco. Permaneceu mais ou menos igual para o de número quatro e subiu um pouco para o de número dois.

Como a inicialização depende de uma parcela dos dados disponíveis, um aumento do número de amostras permite uma inicialização com uma massa de dados maior. Assim, quaisquer erros existentes nos dados de inicialização possuem uma menor contribuição quando o número de amostras é maior. Qualquer *outlier* presente nestes dados teria uma contribuição menor na inicialização se o número de amostras fosse aumentado.

Fazendo uma analogia com o critério de *Nyquist* [12], que diz a freqüência de amostragem de um sinal analógico, para que possa posteriormente ser reconstituído com o mínimo de perda de informação, deve ser igual ou maior a duas vezes a maior freqüência do espectro desse sinal, pode-se dizer que pelo fato do problema pertencer ao domínio discreto quanto maior for o número de amostras disponíveis mais próximo do domínio contínuo estará o "sinal", que no caso corresponde a série temporal das marcações. Assim, a maior freqüência é igual ao inverso do menor tempo de amostragem.

Uma outra linha de raciocínio seria a de diminuir a taxa de amostragem, para aumentar a quantidade de pontos, e portanto, no limite, imaginar que um segmento de reta contínuo aproximaria a verdadeira trajetória do contato, possibilitando que seja efetuada a correta estimativa da posição do mesmo.

De forma a trabalhar mais próximo da realidade que normalmente é encontrada nos sistemas de sensores existentes, onde os dados estão disponíveis no barramento a cada 1s ou 2s, o intervalo de amostragem utilizado a partir de agora será de 2s, que é o pior caso. No entanto, foi mantido o tempo total de amostragem (janela de tempo) igual a 240s, de forma que o número de amostras passará a ser de 120 em cada pernada de 4 minutos. Deste modo um conjunto de 12 dados será empregado para inicializar os algoritmos PL.

O trabalho já estudado que usa o algoritmo EKELUND [1] utilizou uma baixa taxa de amostragem, no caso de 12s, pois trabalhava com dados amostrados segundo medições visuais (óticas).

Cenário	*Marcações iniciais e finais das pernadas*	*Dmin*		D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*				*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
						X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	D_{PL-MOD}	$RMSE_{PL}$	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	***jardas***	**NM**	***jardas***	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
B1	1º pernada [10,1 a 6,3] 2º pernada [6,3 a 7,7]	10411,0	5,14	25310,0 Simul 25418,1	12,50 Simul 12,55	$X_T = 1,359$ $X_O = -0,402$	$Y_T = 13,60$ $Y_O = 0,599$	13,119	0,690	-0,818	1,789	14,358	1,358	13,600	13,120	0,00204
B2	1º pernada [10,1 a 7,9] 2º pernada [7,9 a 8,9]	20360,0	10,05	38294,0 Simul 39108,6	18,91 Simul 19,31	$X_T = 2,667$ $X_O = -0,402$	$Y_T = 20,27$ $Y_O = 0,599$	19,909	0,428	-2,583	3,288	22,867	2,668	20,280	19,919	0,00418
B3	1º pernada [10,5 a 8,9] 2º pernada [8,9 a 9,6]	45813,0	22,61	54080,0 Simul 55613,6	26,70 Simul 27,46	$X_T = 4,267$ $X_O = -0,402$	$Y_T = 28,27$ $Y_O = 0,599$	28,062	0,402	-1,965	2,687	30,412	4,269	28,285	28,078	0,0091
B4	1º pernada [10,2 a 8,7] 2º pernada [8,7 a 8,9]	45813,0	22,61	79739,0 Simul 79733,3	39,37 Simul 39,37	$X_T = 5,73$ $X_O = -0,402$	$Y_T = 39,73$ $Y_O = 0,599$	39,609	1,062	3,201	2,987	37,054	5,705	39,570	39,447	0,1182
B5	1º pernada [0,3 a 358,6] 2º pernada [358,6 a 0,8]	26033,0	12,85	34513,0 Simul 34150,2	17,04 Simul 16,86	$X_T = -0,1667$ $X_O = -0,402$	$Y_T = 17,46$ $Y_O = 0,599$	16,871	0,306	9,123	8,553	8,458	-0.167	17,461	16,864	0,00046

Tabela 6.10: Resumo dos testes efetuados para os cenários "B1" a "B5" utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

6.3. Simulação do Cenário “A" utilizando os Algoritmo PL Modificado

A figura 6.10 representa o movimento original do contato e do navio observador conforme foi proposto em [5], mas agora realizando a estimativa com o uso do algoritmo PL Modificado.

Na figura 6.11, encontram-se as ampliações do resultado, para melhor entendimento, da simulação com a cinemática do trabalho BO-TMA. Encontram-se identificadas pelos respectivos símbolos de ‘+’ e ‘*’ e circundados por círculos, duas posições do contato, sendo uma real e a outra estimada.

A tabela 6.11 apresenta um resumo dos testes efetuados para o cenário “A” utilizando-se os algoritmos EKELUND e PL Modificado. Já na tabela 6.12, estão resumidas as condições inicias para o problema simulado.

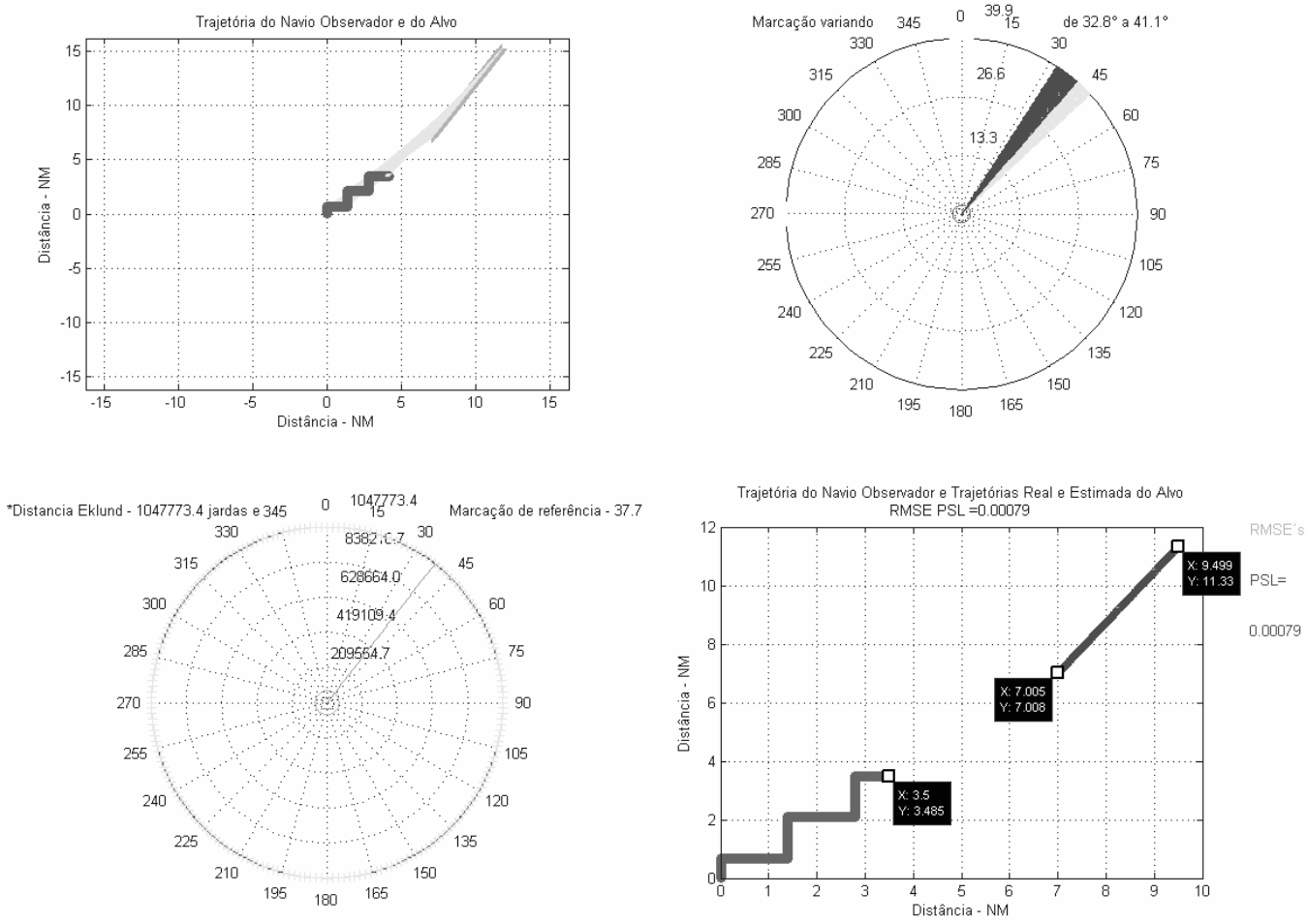


Figura 6.10: Resultado da Simulação com Cenário “A” utilizando o Algoritmo PL Modificado

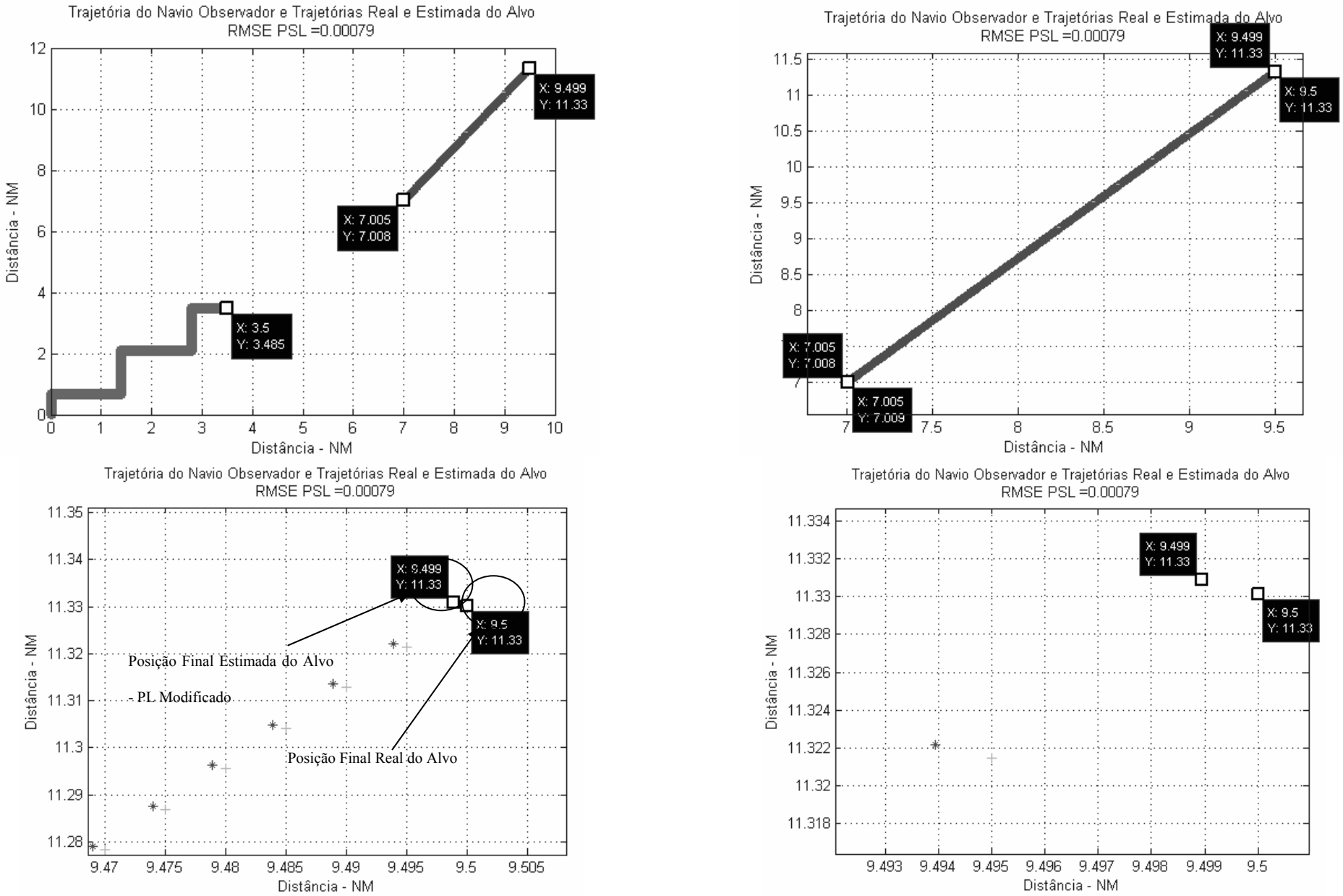


Figura 6.11: Ampliações do Resultado da Simulação com Cenário "A" utilizando o Algoritmo PL Modificado

D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Modificado*			
		X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	$D_{PL\text{-}MOD}$	$RMSE_{PL}$
jardas	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
1047773,4	517,3	$X_T = 9.5$ $X_O = 3,5$	$Y_T = 11,33$ $Y_O = 3,485$	9,87644	9,499	11,33	9,87583	0,00079

Tabela 6.11: Resumo dos testes efetuados para o Cenário "A" utilizando o Algoritmo PL Modificado.

Posição inicial do navio observador X,Y - (NM)	(0 , 0)
Rumo inicial do navio observador na primeira pernada (graus)	000
Instantes de tempo em que o navio observador muda seu rumo (guinada) de 90 graus	$t = 50 + 100 \cdot i$ $i = 0, 1, 2, 3, ...$
Velocidade do navio observador na primeira pernada (nós)	25,4
Posição inicial do contato X,Y - (NM) – para o simulador	(7 , 7)
Rumo do contato na primeira e segunda pernadas (graus)	030
Velocidade do contato na primeira e segunda pernada (nós)	18,0
Intervalo de tempo amostragem em segundos (s)	2
Tempo total de amostragem (Janela de tempo) em segundos (s)	1000
Rumo do navio observador na segunda pernada	090
Velocidade do navio observador na segunda pernada	5,0

Tabela 6.12: Resumo das Condições Iniciais para o Cenário "A" utilizando o Algoritmo PL Modificado.

Pode-se observar na tabela 6.11 que os resultados alcançados estão praticamente iguais aos alcançados na tabela 6.1. Os valores encontrados para a distância final do contato e para o erro RMSE são compatíveis com os resultados alcançado no trabalho original [5], que foi próximo de zero após cerca de 200 iterações, flutuando em torno deste valor até o número final de 500 iterações, conforme fig 3 do citado trabalho.

Os resultados de EKELUND apresentam novamente um erro grande, pois o rumo e velocidades na primeira pernada não foram constantes, devido ao *zig-zag* do navio observador.

6.4. Simulação dos Cenários “C", “D", e “E" utilizando os Algoritmos EKELUND, PL Original e PL Modificado.

6.4.1. Simulação do Cenário “C".

Estão resumidas na tabela 6.13 as condições inicias que são inseridas no simulador MRU a fim de seja gerado o cenário “C” que permite submeter os algoritmos a um conjunto de dados diferente dos anteriormente utilizados.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
C1	0	0	000	13,0	7,0	7,0	030	9,0	090	9,0
C2	0	0	120	5,0	1,0	-9,0	248	16,0	200	17,0
C3	0	0	000	8,0	10,0	0,5	270	10,0	180	9,0
C4	0	0	100	11,0	20,0	1,0	170	12,0	330	9,0
C5	0	0	017	2,0	17,0	2,0	100	15,0	100	20,0

Tabela 6.13: Resumo das condições iniciais para o simulador MRU gerar os cenários “C1” a “C5” - Algoritmos EKELUND, PL Original e PL Modificado.

Os gráficos ampliados da primeira e da quinta corridas com os resultados das simulações são mostrados nas figuras 6.12 e 6.13, para melhor entendimento. As simulações foram feitas utilizando tanto o algoritmo PL Original, quanto o algoritmo PL Modificado, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem e o tempo total de amostragem (janela de tempo) foram iguais a 2s e 240s, respectivamente, nas simulações realizadas.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo modificado foi muito menor do que o erro obtido para o algoritmo original. A tabela 6.14 apresenta um resumo dos testes efetuados para as corridas.

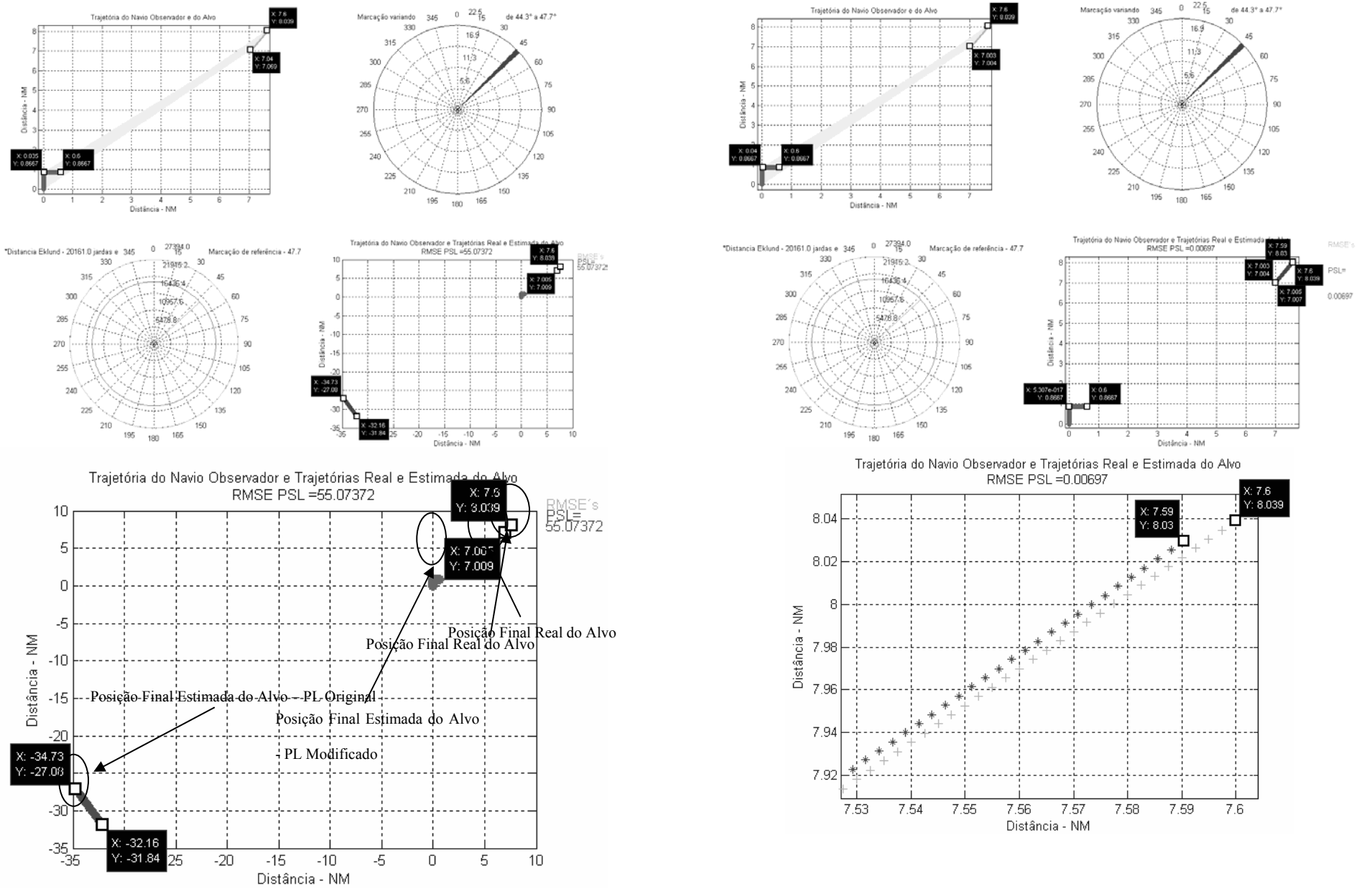


Figura 6.12: Resultado da Simulação para o cenário "C1" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

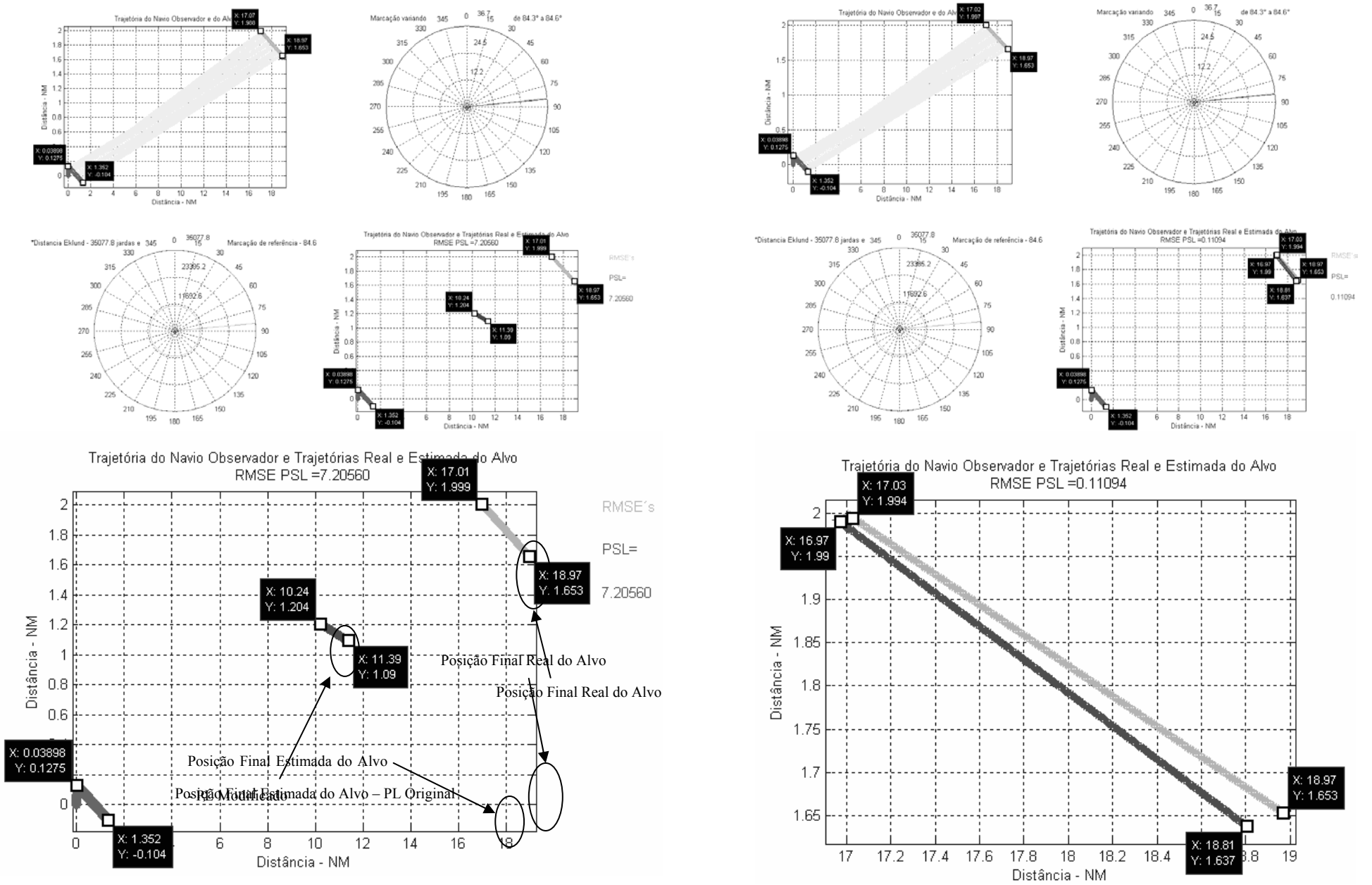


Figura 6.13: Resultado da Simulação para o cenário "C5" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

Pode-se observar na tabela 6.14 que os resultados alcançados pelo algoritmo estimador PL Modificado em termos da distância final estimada do contato, para todos os conjuntos de dados, estão muito próximos da distância final real do contato.

No entanto, o algoritmo PL Original apresentou erros $RMSE_{PL}$ muito grandes, não só com relação às distâncias estimadas, mas também com relação aos rumos estimados e as velocidades estimadas do contato, como na corrida um, uma vez que, apesar da trajetória estimada pertencer à família de possíveis soluções do problema, o contato estaria navegando com o rumo defasado da ordem de 090º e velocidade maior do que a real.

A observação de que o erro $RMSE_{PL\text{-}MOD}$ foi praticamente igual a zero, e o fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real e também da estimativa alcançada pelo algoritmo PL Modificado, demonstra que o mesmo funcionou corretamente em todos os casos.

Cenário	*Marcações iniciais e finais das pernadas*	D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*				*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
				X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	D_{PL-MOD}	$RMSE_{PL}$	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	*jardas*	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
C1	1º pernada [45,0 a 47,7] 2º pernada [47,7 a 44,3]	20161,0	9,954	X_T = 7,600 X_O = 0,600	Y_T = 8,039 Y_O = 0,867	10,022	-34,734	-27,078	45,049	55,074	7,591	8,030	10,009	0,0070
C2	1º pernada [173,7 a 181,7] 2º pernada [181,7 a 185,9]	17698,0	8,734	X_T = -0,978 X_O = -0,099	Y_T = -9,80 Y_O = -1,23	8,615	0,208	-2,580	1,383	7,068	-0,975	-9,783	8,596	0,0144
C3	1º pernada [87,1 a 90,2] 2º pernada [90,2 a 86,3]	18753,7	9,259	X_T = 8,667 X_O = 6,9397e-017	Y_T = 0,50 Y_O =-0,067	8,6855	-4,849	1,384	5,062	14,579	8,651	0,499	8,669	0,0082
C4	1º pernada [87,1 a 89,0] [2º pernada [89,0 a 92,8]	40026,2	19,76	X_T = 20,278 X_O = 0,422	Y_T = -0,58 Y_O = 0,392	19,88	2,577	-0,274	2,2550	18,2578	20,238	-0,574	19,840	0,0518
C5	1º pernada [83,3a 84,6] 2º pernada [84,6 a 84,3]	35077,8	17,32	X_T = 18,97 X_O = 1,352	Y_T = 1,653 Y_O = - 0,10	17,705	11,393	1,0898	10,112	7,206	18,808	1,637	17,542	0,1109

Tabela 6.14: Resumo dos testes efetuados para os cenários "C1" a "C5" utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

6.4.2. Simulação do Cenário "D".

Estão resumidas na tabela 6.15 as condições inicias que são inseridas no simulador MRU a fim de seja gerado o cenário "D" diferente dos outros cenários que foram simulados anteriormente, principalmente com relação a posição inicial do navio observador, que é diferente da origem (0,0).

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
D1	1	2	063	8,0	12,0	50,0	330	11,0	011	13,0
D2	25	47	078	5,0	63,0	15,0	045	12,0	045	12,0
D3	-12	45	265	7,0	65,0	-15,0	024	12,0	127	9,0
D4	1200	1500	254	10,0	1000,0	1510,0	056	14,0	301	9,0
D5	-5	-17	105	15,0	0,5	78,0	235	10,5	014	13,9

Tabela 6.15: Resumo das condições iniciais para o simulador MRU gerar os cenários "D1" a "D5" - Algoritmos EKELUND, PL Original e PL Modificado.

Os gráficos ampliados do terceiro e do quinto conjunto de dados com os resultados das simulações são mostrados nas figuras 6.14 e 6.15, para melhor entendimento. As simulações foram feitas utilizando tanto o algoritmo PL Original, quanto o algoritmo PL Modificado, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem e o tempo total de amostragem (janela de tempo) foram iguais a 2s e 240s, respectivamente, nas simulações realizadas.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo original foi muito maior, acima de 20 vezes e chegando a mais de 15000 vezes, do que o erro obtido para o algoritmo modificado. A tabela 6.16 apresenta um resumo dos testes efetuados para o os conjuntos de dados, com posição inicial do navio observador diferente da origem (0,0).

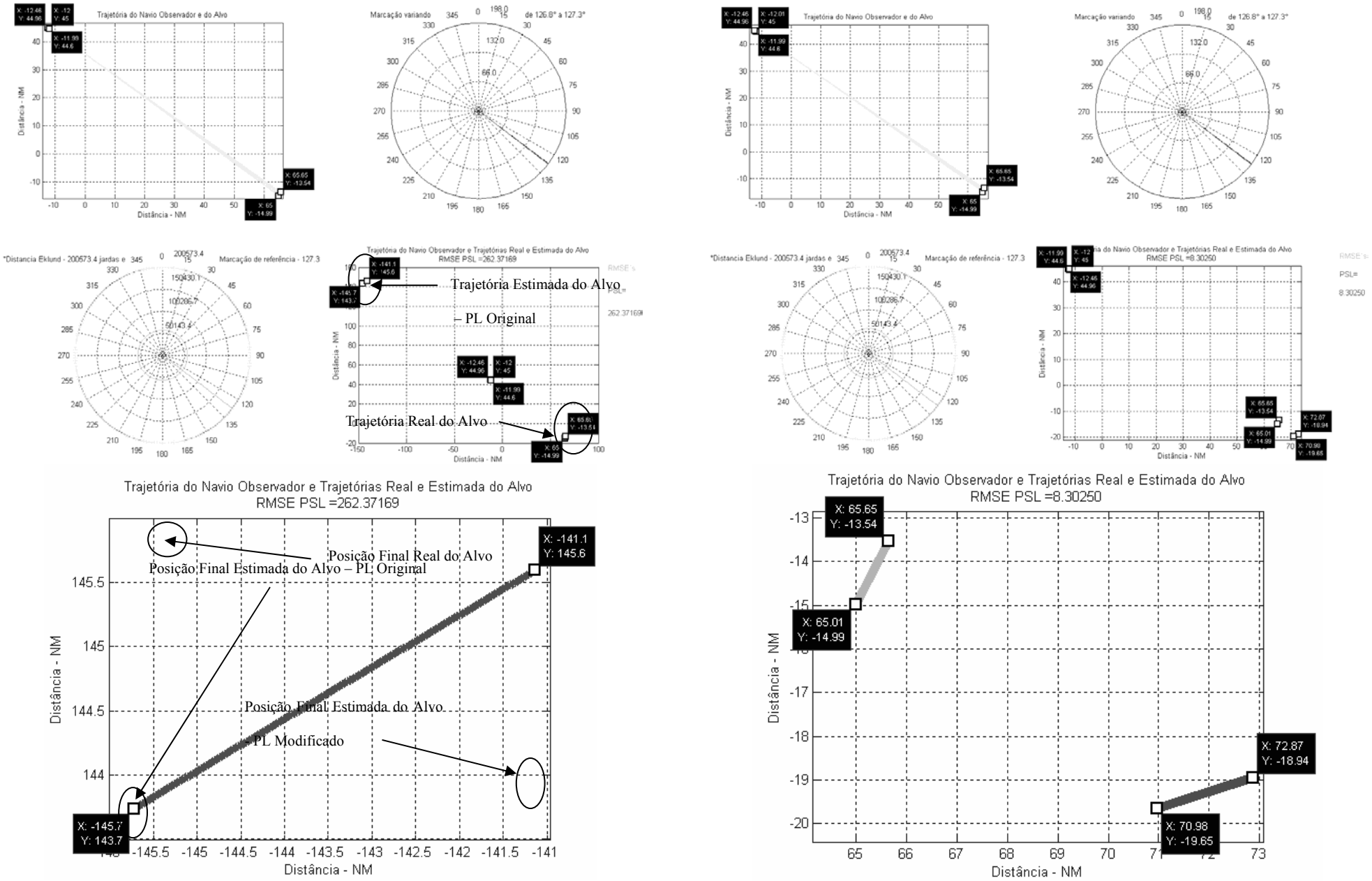


Figura 6.14: Resultado da Simulação para o cenário "D3" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

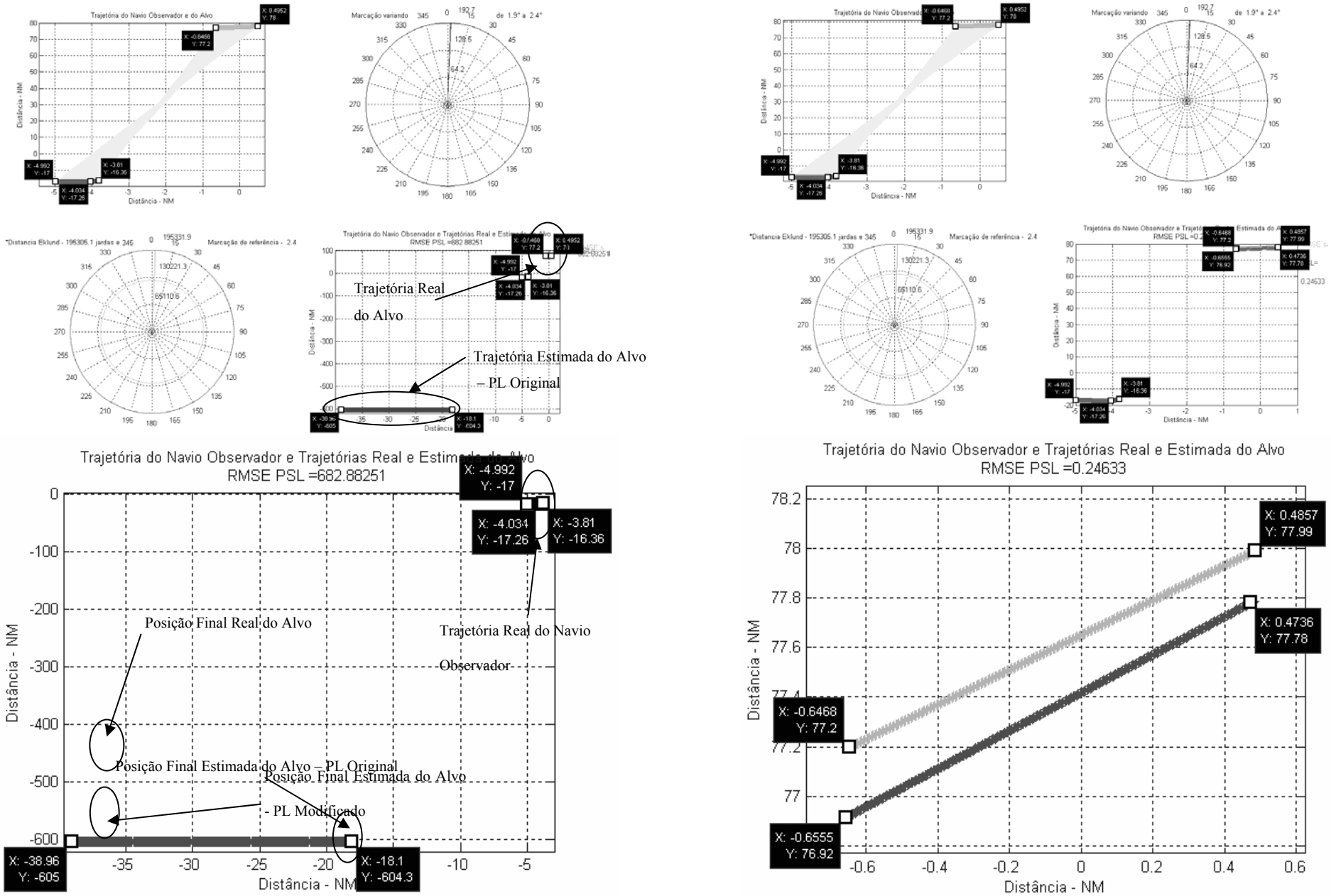


Figura 6.15: Resultado da Simulação para o cenário "D5" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

Pode-se observar na tabela 6.16 que os resultados alcançados pelo algoritmo estimador PL Modificado em termos da distância final estimada do contato, para todos os conjuntos de dados, estão muito próximos da distância final real do contato.

No entanto, o algoritmo PL Original apresentou erros $RMSE_{PL}$ muito grandes, não só com relação às distâncias estimadas, mas também com relação aos rumos estimados e as velocidades estimadas do contato, uma vez que, apesar da trajetória estimada pertencer à família de possíveis soluções do problema, o contato estaria navegando muitas vezes com rumo invertido de quase 180º e velocidade muito maior do que a real. Em alguns casos, como no quinto conjunto de dados, a velocidade do contato seria de aproximadamente 313 nós.

Apesar do erro $RMSE_{PL\text{-}MOD}$ ter sido maior do que o encontrado no caso anterior, quando a posição inicial do navio observador é igual à origem, o fato de que o algoritmo EKELUND ter apresentado uma estimativa próxima da real e também da estimativa alcançada pelo algoritmo PL Modificado, demonstra que o mesmo funcionou corretamente em todos os casos.

No entanto, o algoritmo PL, tanto o Original quanto o Modificado, apresentam uma sensibilidade maior quando são inseridas no mesmo as posições reais absolutas do navio observador. Uma maneira de contornar este problema é a utilização das coordenadas relativas do mesmo, considerando o início da aquisição dos dados como origem e fazendo as posições subseqüentes relativas a mesma.

De todo modo, o algoritmo PL Modificado apresentou um desempenho muito melhor do que o PL Original.

A seguir, serão analisados os comportamentos e os desempenhos dos dois algoritmos com relação a possíveis contatos com características de movimento de aeronaves, i.e., de alvos aéreos, que poderiam ser acompanhados por sistemas MAGE, e que possuem velocidades muito mais altas do que os alvos de superfície ou submarinos.

Cenário	*Marcações iniciais e finais das pernadas*	D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*				*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
				X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	D_{PL-MOD}	$RMSE_{PL}$	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	*jardas*	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
D1	1º pernada [12,9 a 11,9] 2º pernada [11,9 a 11,3]	98278,8	48,52	$X_T = 11,27$ $X_O = 1,64$	$Y_T = 51,27$ $Y_O = 3,09$	49,13	-2890,0	-15158,0	15435,0	15477,0	11,093	50,397	48,239	0,774
D2	1º pernada [130,1 a 129,5] 2º pernada [129,5 a 129,5]	100400,5	49,57	$X_T = 64,13$ $X_O = 25,89$	$Y_T = 16,13$ $Y_O = 47,64$	49,55	-513,8	481,9	692,7	743,6	63,842	16,369	49,171	0,282
D3	1º pernada [127,9 a 127,3] 2º pernada [127,3 a 126,8]	200573,4	99,03	$X_T = 65,65$ $X_O = -11,99$	$Y_T = -13,54$ $Y_O = 44,59$	96,986	-145,7	143,7	166,5	262,4	72,8674	-18,943	106,0073	8,3025
D4	1º pernada [272,9 a 273,1] 2º pernada [273,1 a 273,2]	407339,5	201,1	$X_T = 1001,50$ $X_O = 1198,80$	$Y_T = 1511,00$ $Y_O = 1500,10$	197,6	1713,5	1469,9	515,6	713,2	991,4256	1511,6	207,7367	8,5967
D5	1º pernada [3,3 a 2,4] 2º pernada [2,4 a 1,9]	195305,1	96,43	$X_T = -0,65$ $X_O = -3,81$	$Y_T = 77,20$ $Y_O = -16,36$	93,61	-18,1	-604,3	588,1	682,9	-0,6555	76,9154	93,3284	0,2463

Tabela 6.16: Resumo dos testes efetuados para os cenários "D1" a "D5" utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

6.4.3. Simulação do Cenário "E".

Estão resumidas na tabela 6.17 as condições inicias que são inseridas no simulador MRU a fim de seja gerado o cenário "E" diferente dos outros cenários que foram simulados até o momento, principalmente com relação às velocidades dos contatos que são muito mais altas do que as velocidades dos contatos de superfície ou submarinos. Foi feita a suposição de que os contatos seriam aéreos e detectados por um equipamento MAGE.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	**NM**	**Graus**	**nós**	**NM**	**NM**	**Graus**	**nós**	**Graus**	**nós**
E1	0	0	040	9,0	0,5	17,0	012	240,0	280	12,0
E2	0	0	040	9,0	2,0	12,0	325	120,0	280	12,0
E3	0	0	000	8,0	-10,0	15,0	270	110,0	330	15,0
E4	0	0	100	11,0	-20,0	-1,0	170	98,0	230	9,5
E5	0	0	317	4,5	17,0	-32,0	275	155,0	130	19,0

Tabela 6.17: Resumo das condições iniciais para o simulador MRU gerar os cenários "E1" a "E5" - Algoritmos EKELUND, PL Original e PL Modificado.

Os gráficos ampliados do primeiro e do terceiro conjunto de dados com os resultados das simulações são mostrados nas figuras 6.16 e 6.17, para melhor entendimento. As simulações foram feitas utilizando tanto o algoritmo PL Original, quanto o algoritmo PL Modificado, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem e o tempo total de amostragem (janela de tempo) foram iguais a 2s e 240s, respectivamente, nas simulações realizadas.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo modificado foi muito menor do que o erro obtido para o algoritmo original. A tabela 6.18 apresenta um resumo dos testes efetuados para o os conjuntos de dados.

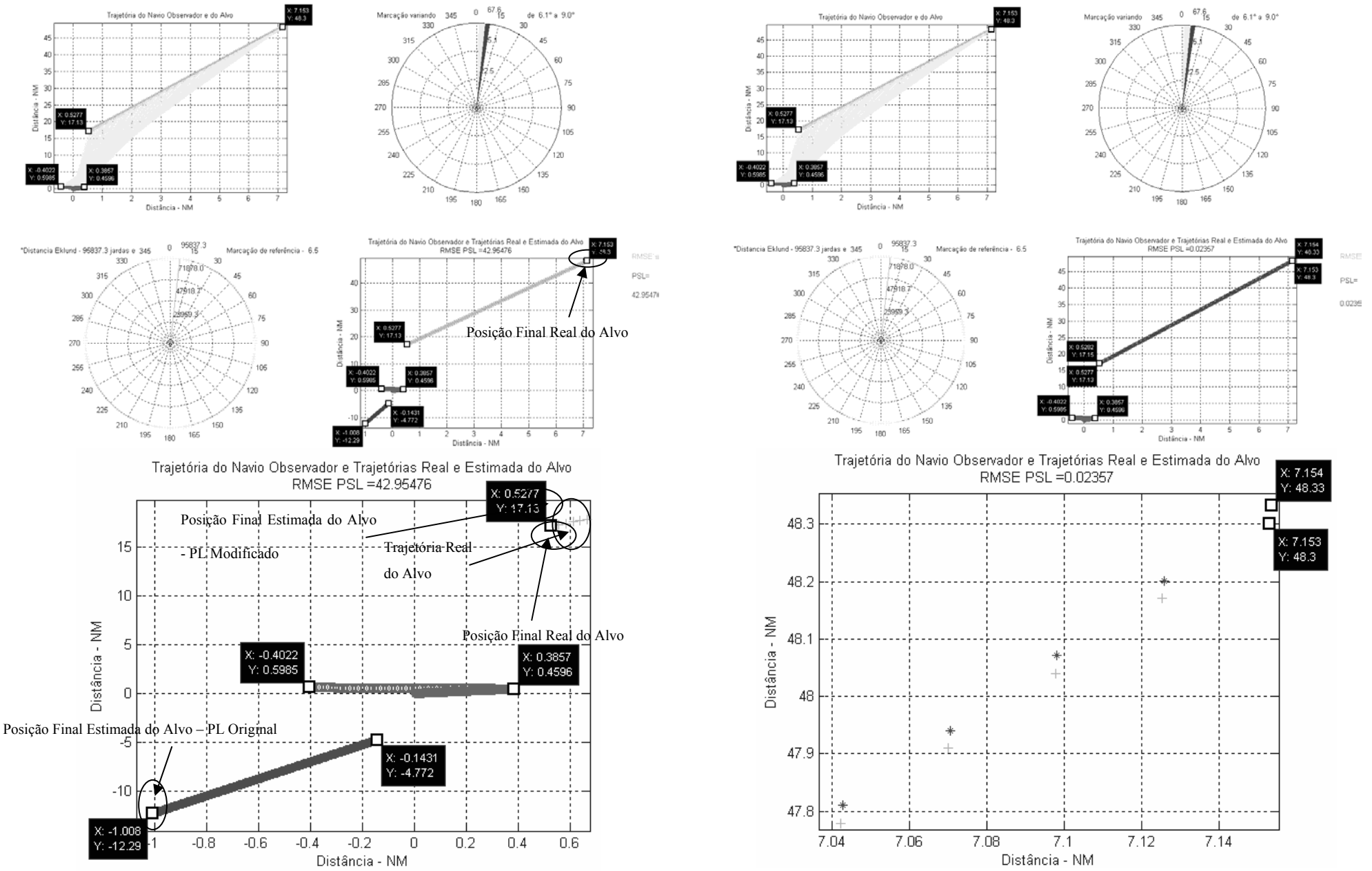


Figura 6.16: Resultado da Simulação para o cenário "E1" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

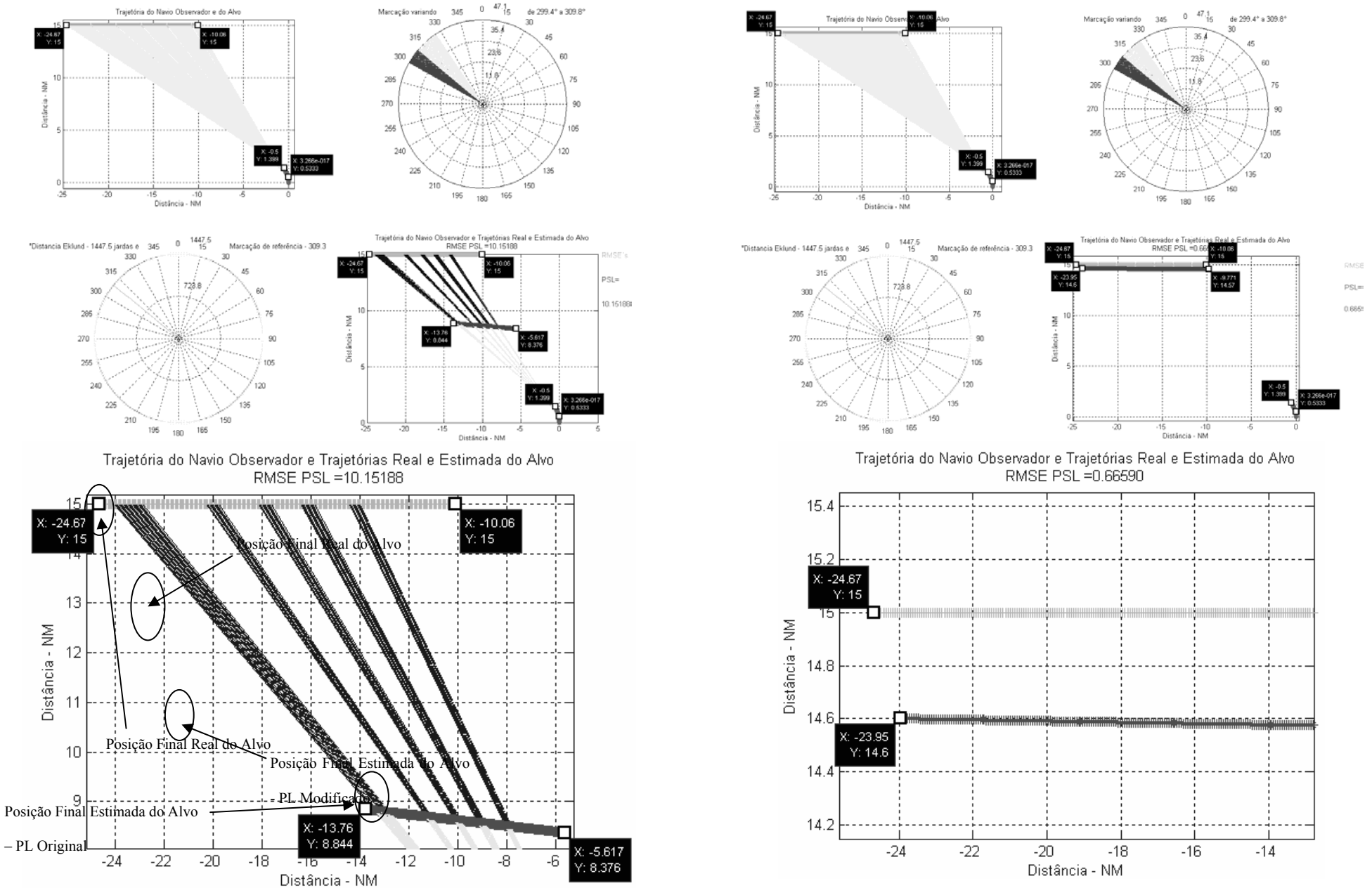


Figura 6.17: Resultado da Simulação para o cenário "E3" utilizando os Algoritmos EKELUND, PL Original e PL Modificado

Pode-se observar na tabela 6.18 que os resultados alcançados pelo algoritmo estimador PL Modificado em termos da distância final estimada do contato, para todos os conjuntos de dados, estão muito próximos da distância final real do contato.

No entanto, o algoritmo PL Original apresentou erros $RMSE_{PL}$ muito grandes, não só com relação às distâncias estimadas, mas também com relação aos rumos estimados e as velocidades estimadas do contato, uma vez que, apesar da trajetória estimada pertencer à família de possíveis soluções do problema, o contato estaria navegando muitas vezes com rumo invertido de quase 180º e velocidade muito destoante do que a real. Em alguns casos, como no primeiro conjunto de dados, a velocidade do contato seria de aproximadamente 60 nós, ou seja, um sexto da velocidade real. Além disso, o rumo estaria invertido de pelo menos 169 graus.

A observação de que o erro $RMSE_{PL\text{-}MOD}$ foi da mesma ordem de grandeza do que o encontrado no item 6.4.1, quando a posição inicial do navio observador é igual à origem e as velocidades são proporcionais ao de um contato de superfície ou submarinos, demonstra que o mesmo funcionou corretamente em todos os casos, apresentando um comportamento robusto.

O fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real somente no caso da primeira corrida, visto que nas demais o erro de estimativa da distância EKELUND foi elevado, deve-se ao fato de que nestes casos, com velocidades elevadas, as marcações sofrem grandes variações com taxas elevadas, fazendo com que a aproximação que considera somente as velocidades transversais não seja mais válida.

Constata-se, portanto que o algoritmo PL Modificado apresentou um desempenho muito melhor do que o PL Original em todas as situações de movimentos envolvendo contatos e o navio observador.

A seguir, serão analisados os comportamentos e os desempenhos dos algoritmos com relação à inserção de ruído, e principalmente como é o desempenho das *wavelets* em uma aplicação de *denoising*, para o caso de TMA.

Cenário	*Marcações iniciais e finais das pernadas*	D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Original [5]*				*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
				X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL}	Y_{PL}	D_{PL-MOD}	$RMSE_{PL}$	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	*jardas*	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
E1	1º pernada [1,7 a 6,1] 2º pernada [6,1 a 9,0]	95837,3	47,32	$X_T = 7,153$ $X_O = -0,402$	$Y_T = 48,301$ $Y_O = 0,599$	48,30	-1,008	-12,292	12,905	42,955	7,154	48,332	48,328	0,0236
E2	1º pernada [9,5 a 350,7] 2º pernada [350,7 a 344,5]	10561,3	5,21	$X_T = -7,177$ $X_O = -0,402$	$Y_T = 25,106$ $Y_O = 0,599$	25,43	-0,187	3,837	3,246	16,614	-7,173	25,112	25,431	0,0049
E3	1º pernada [326,3 a 309,8] 2º pernada [309,8 a 299,4]	1447,5	0,71	$X_T = -24,667$ $X_O = -0,500$	$Y_T = 15,000$ $Y_O = 1,399$	27,73	-13,764	8,844	15,210	10,152	-23,947	14,600	26,908	0,6659
E4	1º pernada [267,1 a 249,5] 2º pernada [249,5 a 233,4]	11671,8	5,76	$X_T = -17,731$ $X_O = 0,237$	$Y_T = -13,868$ $Y_O = -0,534$	22,38	6,250	10,603	12,389	31,363	-17,897	-13,990	22,581	0,1879
E5	1º pernada [152,0 a 167,6] 2º pernada [167,6 a 188,4]	22807,4	11,26	$X_T = -3,588$ $X_O = 0,766$	$Y_T = -30,199$ $Y_O = -0,595$	29,92	-0,580	-1,219	1,484	30,837	-3,573	-30,173	29,895	0,0262

Tabela 6.18: Resumo dos testes efetuados para os cenários "E1" a "E5" utilizando os Algoritmos EKELUND, PL e PL Modificado com intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

6.5. *Wavelets Denoising* empregada na redução de ruído em problemas de análise de movimento de alvos - TMA.

É inerente a qualquer processo de aquisição de dados a existência de ruído, mesmo que sejam adotados diversos procedimentos e precauções, tanto ao nível do *hardware* quanto em termos do *software* que se encontram embarcados na maioria dos sistemas de instrumentação eletrônica ou de processamento de sinais.

Uma medida que esteja contaminada com um nível de ruído muito grande, pode provocar problemas de convergência de um algoritmo ou até mesmo provocar falhas de operação de sistemas. Na medida do possível, os projetistas procuram tornar os sistemas eletrônicos robustos às interferências eletromagnéticas. Com o avanço das tecnologias, pode-se nos dias atuais utilizar sinais eletrônicos diferencias, em sistemas de comunicação, ou ainda filtrar através de circuitos analógicos e digitais, com o uso de processadores digitais de sinais (DSP), ruídos cujas componentes principais de freqüência são conhecidas ou não.

Uma vertente que se encontra muito em voga nos dias atuais no processamento digital de sinais [10] é o uso das *Wavelets* e das Transformadas Discretas de *Wavelets* (DWT) como ferramenta na redução de ruído, i.e. em *Denoising,* tanto para o ruído branco aditivo, como para um ruído com componentes de freqüências cujos coeficientes podem ser determinados.

Nos sistemas de sensores existentes em Navios ou Aeronaves, devido a quantidade cada vez maior de fontes de ruído que são encontradas não só a nível interno, como também externo, estes cuidados também devem ser tomados no projeto.

No caso em estudo, onde se deseja no futuro desenvolver uma ferramenta confiável de estimação da posição de contatos, foram utilizadas as *Wavelets* para reduzir o ruído presente na série temporal de marcações, de forma independente de qualquer outro tipo de tratamento ou filtragem do sinal que possa existir anteriormente.

Um ruído gaussiano de 1% foi aleatoriamente somado aos dados simulados de Latitude e Longitude obtidos da simulação de um Movimento Retilíneo Uniforme (MRU) do contato.

A seguir, são apresentadas as tabelas 6.19 a 6.22 que resumem os testes que foram feitos com as *Wavelets* empregadas em *Denoising* para os conjuntos de dados referentes ao cenário “A”.

6.5.1. *Wavelet Denoising* utilizando o Algoritmo PL Modificado no Cenário "A".

A tabela 6.19 sintetiza os experimentos realizados com as *Wavelets Symmlet 8 ('sym8')* para o cenário "A", empregando o Algoritmo PL Modificado. Neste caso, foram efetuadas medidas com relação ao sinal sem ruído gerado pelo simulador MRU, e diversos parâmetros foram alterados na função *wden* de forma a tentar encontrar o melhor ajuste para as simulações do trabalho BO-TMA.

'sym8'			***Regra de threshold: 'sqtwolog'***					
			Fator multiplicativo de re-escalamento do threshold					
Nível	***RMSE da Latitude c/ ruído adicionado***	***RMSE da Longitude c/ ruído adicionado***	***'one'***		***'sln'***		***'mln'***	
			RMSE da Latitude após o denoising	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***
2	0,018854	0,019999	0,009994	0,009435	0,009994	0,009443	0,009994	0,009437
3	0,018854	0,019999	0,006936	0,007151	0,006933	0,007152	0,006936	0,007152
4	0,018854	0,019999	0,006757	0,005461	0,00643	0,00535	0,006473	0,005438
5	0,018854	0,019999	0,007681	0,005145	0,007158	0,004944	0,00724	0,005106
6	0,018854	0,019999	0,013544	0,008875	0,007164	0,006988	0,011925	0,008849
7	0,018854	0,019999	0,026216	0,016479	0,008864	0,009135	0,07874	0,07874
8	0,018854	0,019999	0,060828	0,036056	0,01063	0,011219	0,06	0,036056
9	0,018854	0,019999	0,126095	0,07874	0,011328	0,011902	0,130767	0,07874

Tabela 6.19: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'sqtwolog'* para o Cenário "A" e o Algoritmo PL Modificado.

A melhor situação alcançada para a regra de *threshold 'sqtwolog'* foi com o nível quatro para todos os fatores de re-escalamento. No caso, o fator multiplicativo de re-escalamento '*sln*' apresentou um desempenho ligeiramente superior, tanto em termos da redução de ruído na Latitude quanto na Longitude, uma vez que esta redução foi da ordem de 2,93 vezes e da ordem de 3,74 vezes, respectivamente.

No caso da tabela 6.20, os experimentos realizados com as *Wavelets Symmlet 8 ('sym8')* o cenário "A", mostram que os níveis de decomposição estão situados entre o quatro e o seis.

'sym8'			***Regra de threshold: 'rigrsure'***					
			Fator multiplicativo de re-escalamento do threshold					
Nível	***RMSE da Latitude c/ ruído adicionado***	***RMSE da Longitude c/ ruído adicionado***	***'one'***		***'sln'***		***'mln'***	
			RMSE da Latitude após o denoising	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***
2	0,0220898	0,0200893	0,011679	0,010275	0,011751	0,01153	0,011843	0,011571
3	0,0220898	0,0200893	0,007355	0,006157	0,008415	0,008849	0,00854	0,008902
4	0,0220898	0,0200893	0,006226	0,00426	0,007218	0,007638	0,007427	0,007699
5	0,0220898	0,0200893	0,00797	0,004105	0,007263	0,007114	0,007641	0,007176
6	0,0220898	0,0200893	0,013888	0,006617	0,007153	0,006966	0,007558	0,007351
7	0,0220898	0,0200893	0,026011	0,013665	0,007419	0,006985	0,017735	0,009203
8	0,0220898	0,0200893	0,061644	0,034641	0,007379	0,007048	0,052915	0,030575
9	0,0220898	0,0200893	0,064807	0,055678	0,011346	0,008392	0,09798	0,053852

Tabela 6.20: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'rigrsure'* para o Cenário "A" e o Algoritmo PL Modificado.

A melhor situação de redução do nível de ruído para a regra de *threshold 'rigrsure'* foi alcançada no nível quatro usando o fator multiplicativo de re-escalamento '*one*'. O mesmo apresentou um desempenho ligeiramente superior, tanto em termos da redução de ruído na Latitude quanto na Longitude, uma vez que esta redução foi da ordem de 3,55 vezes e da ordem de 4,72 vezes, respectivamente.

No caso da Longitude, esta redução de 4,72 vezes foi a melhor dentre todas as simulações realizadas. Portanto, este ajuste de parâmetros poderia ser um possível candidato para a ferramenta de redução de ruído com *Wavelet Denoising.*

Para a tabela 6.21, que sintetiza os experimentos realizados para o cenário "A" com as *Wavelets Symmlet 8 ('sym8')* no caso da *Regra de threshold 'heursure'*, os melhores resultados novamente estão situados nos níveis quatro, com dois melhores casos, e seis, que obteve uma situação de vantagem.

'sym8'			***Regra de threshold: 'heursure'***					
			Fator multiplicativo de re-escalamento do threshold					
			'one'		***'sln'***		***'mln'***	
Nível	***RMSE da Latitude c/ ruído adicionado***	***RMSE da Longitude c/ ruído adicionado***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***
2	0,0207299	0,0209239	0,009264	0,010906	0,009282	0,010906	0,009282	0,010906
3	0,0207299	0,0209239	0,006837	0,007768	0,006862	0,007768	0,006862	0,007768
4	0,0207299	0,0209239	0,005443	0,004768	0,004772	0,004502	0,005416	0,004763
5	0,0207299	0,0209239	0,006817	0,004932	0,004602	0,004836	0,006645	0,00483
6	0,0207299	0,0209239	0,013595	0,006862	0,00462	0,00459	0,007873	0,005791
7	0,0207299	0,0209239	0,026226	0,013422	0,017828	0,024426	0,023138	0,012481
8	0,0207299	0,0209239	0,061644	0,034641	0,004861	0,004444	0,059161	0,033166
9	0,0207299	0,0209239	0,064031	0,056569	0,004961	0,005652	0,130384	0,072801

Tabela 6.21: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'heursure'* para o Cenário "A" e o Algoritmo PL Modificado.

A melhor situação absoluta alcançada, para a regra de *threshold 'heursure'*, foi com o nível seis para o fator multiplicativo de re-escalamento 'sln'. O mesmo apresentou um desempenho ligeiramente superior, tanto em termos da redução de ruído na Latitude, que foi da ordem de 4,49 vezes, quanto na Longitude, que foi da ordem de 4,56 vezes.

A redução de 4,72 vezes alcançada para a Latitude foi a melhor para as simulações realizadas. Novamente, este ajuste de parâmetros poderia ser uma opção a ser empregado na redução de ruído com *Wavelet Denoising*.

Finalmente, a tabela 6.22 sintetiza o último experimento realizado com as *Symmlets 8 ('sym8')* para a redução de ruído que foi adicionado ao conjunto de dados referente ao cenário "A", empregando o Algoritmo PL Modificado, ao se utilizar a *Regra de threshold 'minimaxi'*.

'sym8'			***Regra de threshold: 'minimaxi'***					
			Fator multiplicativo de re-escalamento do threshold					
Nível	***RMSE da Latitude c/ ruído adicionado***	***RMSE da Longitude c/ ruído adicionado***	***'one'***		***'sln'***		***'mln'***	
			RMSE da Latitude após o denoising	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***	***RMSE da Latitude após o denoising***	***RMSE da Longitude após o denoising***
2	0,0202307	0,0213282	0,010321	0,011713	0,010321	0,011713	0,010321	0,011713
3	0,0202307	0,0213282	0,007594	0,007709	0,007594	0,007709	0,007594	0,007709
4	0,0202307	0,0213282	0,005715	0,006532	0,005643	0,006534	0,005715	0,006532
5	0,0202307	0,0213282	0,006904	0,006126	0,006459	0,006128	0,006904	0,006126
6	0,0202307	0,0213282	0,01407	0,008501	0,008175	0,006814	0,01407	0,008501
7	0,0202307	0,0213282	0,02681	0,0157	0,010195	0,008849	0,02681	0,0157
8	0,0202307	0,0213282	0,061644	0,036056	0,01204	0,010869	0,061644	0,036056
9	0,0202307	0,0213282	0,128062	0,07746	0,01286	0,011715	0,133791	0,07746

Tabela 6.21: Resumo dos testes com *Wavelet Denoising* utilizando a regra de *threshold 'minimaxi'* para o Cenário "A" e o Algoritmo PL Modificado.

A melhor situação alcançada para a regra de *threshold 'minimaxi'* foi com o nível quatro para os fatores de re-escalamento *'one' e 'mln'*. Os mesmos obtiveram uma redução da ordem de 3,54 vezes e da ordem de 3,27 vezes, para a Latitude e para a Longitude, respectivamente.

No caso, o fator multiplicativo de re-escalamento *'sln'* apresentou um desempenho ligeiramente inferior aos outros dois, tanto em termos da redução de ruído na Latitude quanto na Longitude, pois ficou com uma redução de 2,47 e 3,13 para as coordenadas citadas.

6.5.2. Resultados do emprego de *Wavelets Denoising* para os cenários "B1" e "B5" utilizando os Algoritmos EKELUND e PL Modificado.

Estão resumidas na tabela 6.23 as condições inicias que são inseridas no simulador MRU a fim de sejam gerados os cenários "B1" a "B5" de modo a executar os testes com *Wavelets Denosing*.

Cenário	*Condições Iniciais do Navio Observador – Primeira Pernada*				*Condições Iniciais do Contato*				*Condições Iniciais do Navio Observador – Segunda Pernada*	
	X_O (0)	Y_O (0)	Rv_O (0)	V_O (0)	X_C (0)	Y_C (0)	Rv_C (0)	V_C (0)	Rv_O (120)	V_O (120)
	NM	NM	Graus	nós	NM	NM	Graus	nós	Graus	nós
B1	0	0	040	9,0	2,2	12,4	325	11,0	280	12,0
B2	0	0	040	9,0	3,4	19,0	330	11,0	280	12,0
B3	0	0	040	9,0	5,0	27,0	330	11,0	280	12,0
B4	0	0	040	9,0	7,0	39,0	300	11,0	280	12,0
B5	0	0	040	9,0	0,1	17,0	330	4,0	280	12,0

Tabela 6.23: Resumo das condições iniciais para o simulador MRU gerar os cenários "B1" a "B5" para testes com *Wavelets Denoising* – Algoritmos EKELUND e PL Modificado.

Os gráficos ampliados do primeiro conjunto de dados com os resultados das simulações usando ruído de 0,5% e 1% são mostrados nas figuras 6.18 e 6.19, respectivamente. As simulações foram feitas utilizando o algoritmo PL Modificado, que conforme foi visto, apresenta desempenho em muito superior ao do algoritmo PL Original, sob o mesmo conjunto de dados e com as mesmas condições iniciais. O intervalo de amostragem e o tempo total de amostragem (janela de tempo) continuam iguais a 2s e 240s, respectivamente, nas simulações realizadas.

Novamente as posições do contato, sendo uma real e a outra estimada, encontram-se identificadas pelos respectivos símbolos de '+' e '*', nas cores vermelho e ciano na tela original.

Pode-se observar por estes gráficos que o erro de estimação para o algoritmo PL Modificado, quando existe ruído aleatório adicionado ao sinal de marcações, foi um pouco maior do que o erro obtido anteriormente para o mesmo algoritmo, quando recebe a série de marcações limpa de interferências. A tabela 6.24 apresenta um resumo dos testes efetuados para o os conjuntos de dados.

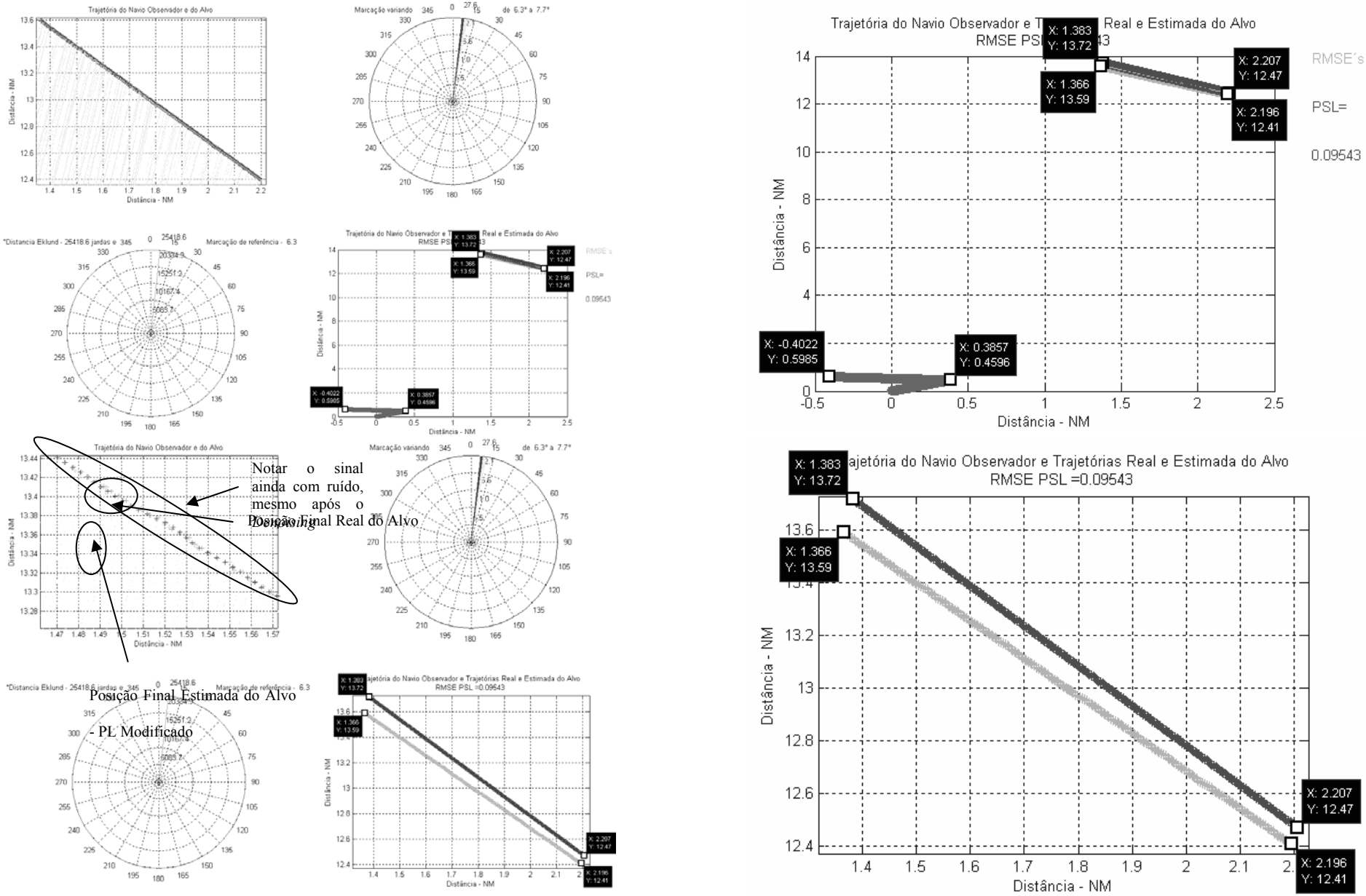


Figura 6.18: : Resultado dos testes com *Wavelets Denoising* para o cenário "B1", nível de ruído adicionado de 0,5%, utilizando os Algoritmos EKELUND e PL Modificado.

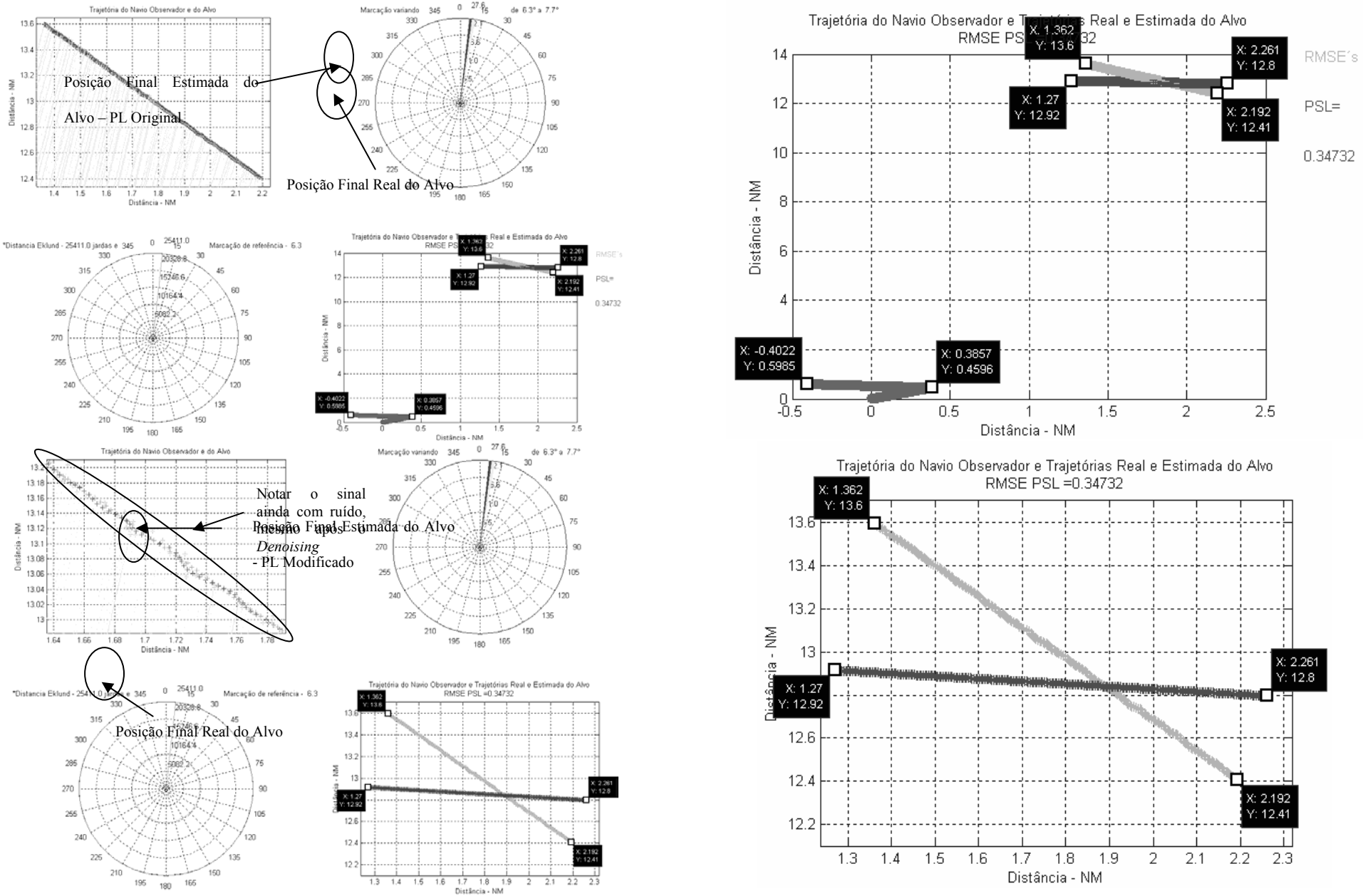


Figura 6.19: Resultado dos testes com *Wavelets Denoising* para o cenário "B1", nível de ruído adicionado de 1,0%, utilizando os Algoritmos EKELUND e PL Modificado.

É possível visualizar que o processo de *Denoising* empregando *Wavelets* não consegue retirar totalmente o ruído que foi adicionado ao sinal temporal de marcações. Se forem aumentados os níveis de detalhamento, na tentativa de eliminar de forma mais acentuada o ruído, provoca-se uma compressão nos dados, principalmente nas extremidades, fato que gera uma distorção indesejável. A mesma prejudica a inicialização do algoritmo PL Modificado, pois conforme já foi visto anteriormente, utiliza-se uma massa de dados inicial de cerca de 10% da quantidade de dados disponíveis, nesse procedimento.

Provavelmente é este fator que prejudica o desempenho do algoritmo, quando se tenta eliminar o ruído completamente.

Conforme se pode observar também, os algoritmos PL, tanto o original que foi proposto [5], como o aqui desenvolvido a partir da modificação do anterior, não são tão imunes ao ruído como afirmam os autores do *paper*. De todo modo, existem outras soluções de filtragem que poderiam ser implementadas e testadas, sendo, portanto citadas no capítulo 8 desta dissertação.

Poderia-se também aumentar a massa de dados disponível, diminuindo-se o intervalo de amostragem de 2s para 1s, mantendo-se o tempo total de amostragem (janela de tempo) em 240s. Com isso, uma quantidade maior de dados estaria disponível para a inicialização do algoritmo.

Pode-se observar na tabela 6.24 que os resultados alcançados pelo algoritmo estimador PL Modificado em termos da distância final estimada do contato, para todos os conjuntos de dados, estão muito próximos da distância final real do contato, e também dos resultados obtidos na tabela 6.10.

Entretanto, se for considerado que todas as linhas de visada, que são as retas suporte a partir das quais se medem as marcações, passam pela posição estimada do contato, chega-se a conclusão que o erro de distância provocaria um erro no cálculo da distância, na solução do tiro, mas não um erro em termos da marcação final estimada. Este fato não seria conveniente, mas é passível de ser controlável uma vez que o armamento necessita de uma semi-esfera de incerteza, dentro da qual o contato estaria localizado. Em muitos casos, os mesmos possuem processamento digital e navegação embarcada, podendo fazer o rastreamento e as correções necessárias.

A observação que o erro $RMSE_{PL\text{-}MOD}$ foi um pouco maior, e o fato de que o algoritmo EKELUND apresentou uma estimativa próxima da real e também da estimativa alcançada pelo algoritmo PL Modificado, demonstra que o mesmo funcionou corretamente em todos os casos, mesmo com ruído somado.

Pretende-se com isso comprovar que o algoritmo PL Modificado possui realmente uma maior robustez do que o PL Original, e que o pré-processamento dos sinais de entrada, utilizando algum tipo de filtragem, como foi o caso com *Wavelet Denoising,* podem auxiliar o sistema na busca da melhor estimativa da posição do contato.

Conjunto de Dados	*Marcações iniciais e finais das pernadas*	D_{EK}		*Posições Finais do Contato e do Navio Observador Calculadas pelo Simulador MRU*			*Posição Final do Contato - Método PL Modificado - Proposto por esta Dissertação*			
				X_{SIMUL}	Y_{SIMUL}	D_{SIMUL}	X_{PL-MOD}	Y_{PL-MOD}	D_{PL-MOD}	$RMSE_{PL-MOD}$
	Graus	***jardas***	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**	**NM**
1	1º pernada [10,1 a 6,3] 2º pernada [6,3 a 7,7]	25310,0	12,50	$X_T = 1,359$ $X_O = -0,402$	$Y_T = 13,60$ $Y_O = 0,599$	13,119	1,270	12,915	12,430	0,3473
		Simul 25418,1	Simul 12,55							
2	1º pernada [10,1 a 7,9] 2º pernada [7,9 a 8,9]	38294,0	18,91	$X_T = 2,667$ $X_O = -0,402$	$Y_T = 20,27$ $Y_O = 0,599$	19,909	1,308	11,517	11,051	5,6185
		Simul 39118,8	Simul 19,31							
3	1º pernada [10,5 a 8,9] 2º pernada [8,9 a 9,6]	54080,0	26,70	$X_T = 4,267$ $X_O = -0,402$	$Y_T = 28,27$ $Y_O = 0,599$	28,062	4,716	30,924	30,754	1,7930
		Simul 55613,6	Simul 27,46							
4	1º pernada [10,2 a 8,7] 2º pernada [8,7 a 8,9]	79739,0	39,37	$X_T = 5,73$ $X_O = -0,402$	$Y_T = 39,73$ $Y_O = 0,599$	39,609	31,919	4,508	31,703	5,5439
		Simul 79733,3	Simul 39,37							
5	1º pernada [0,3 a 358,6] 2º pernada [358,6 a 0,8]	34513,0	17,04	$X_T = -0,1667$ $X_O = -0,402$	$Y_T = 17,46$ $Y_O = 0,599$	16,871	-0,200	14,898	14,301	1,3354
		Simul 34150,2	Simul 16,86							

Tabela 6.24: Resultados do emprego de *Wavelets Denoising* para o cenário "B1", nível de ruído adicionado de 1,0%, utilizando os Algoritmos EKELUND e PL Modificado.

Os resultados que estão condensados na tabela acima mostram que o ruído presente nas amostras iniciais prejudica o processo de inicialização do algoritmo PL Modificado. Mesmo com o uso da técnica de *denoising* com o uso das *wavelets*, os resultados apresentam alguma deterioração se comparados aos resultados cujos dados estão livres de ruído.

6.6. Tempo de Execução dos Algoritmos EKELUND, PL Original e PL Modificado em um mesmo computador.

As Tabelas a seguir, da 6.25 a 6.30 apresentam o tempo médio de execução medido em segundos dos algoritmos EKELUND, PL Original e PL Modificado para todos os conjuntos de dados e corridas para os quais foram efetuados os testes.

Foi utilizada, para a realização dos experimentos, uma plataforma com a seguinte especificação:

- CPU AMD Sempron™ 1.61 GHz, com 512 MB de memória RAM, e 250 GB de "*Hard Disk"* particionado com três partições, sendo duas de 83GB formatadas NTFS para o sistema operacional *Windows XP Professional version 2002 - Service Pack 2* e a partição restante de cerca de 68GB formatada pelo com sistema operacional *Linux*, empregando a distribuição *OpenSUSE 10.3*;
- Placa Mãe GIGABYTE GA-K8 NS Pro para processador AMD; e
- Sistema Operacional: *Windows XP Professional version 2002 - Service Pack 2.*

A tabela 6.25 compila os resultados do tempo médio de processamento em segundos dos algoritmos em estudo:

Tabela de Simulações	***Algoritmo***	***Tempo Médio em segundos***	***Convergência do Algoritmo***
Tabela 6.1 – Cenário "A"	***EKELUND***	1,0215	***Não se aplica***
	PL Original	0,9414	***Converge***
Tabela 6.11 - Cenário "A"	***EKELUND***	1,0215	***Não se aplica***
	PL Modificado	9,7440	***Converge***

Tabela 6.25: Tempo de execução utilizando Cenário "A" utilizando os Algoritmos PL Original e PL Modificado.

Apesar do tempo de processamento ser, aparentemente, elevado, o algoritmo PL Modificado mostrou-se bastante estável e razoavelmente rápido, se comparado ao outro algoritmo PL que não converge como visto em todas as situações estudadas. Já o EKELUND serve como uma ferramenta auxiliar na tomada de decisão.

A compilação dos resultados do tempo médio de processamento, em segundos, dos algoritmos em estudo estão na tabela 6.26, que se encontra abaixo:

Conjunto de Dados da Tabela 6.8 - Cenário "B"	*Algoritmo*	*Tempo Médio em segundos*	*Convergência do Algoritmo*
B1	***EKELUND***	0,22	***Não se aplica***
	PL Original	0,21	***Não converge***
	PL Modificado	0,77	***Converge***
B2	***EKELUND***	0,22	***Não se aplica***
	PL Original	0,23	***Não converge***
	PL Modificado	1,17	***Converge***
B3	***EKELUND***	0,25	***Não se aplica***
	PL Original	0,23	***Não converge***
	PL Modificado	0,77	***Converge***
B4	***EKELUND***	0,23	***Não se aplica***
	PL Original	0,23	***Não converge***
	PL Modificado	0,64	***Converge***
B5	***EKELUND***	0,46	***Não se aplica***
	PL Original	0,23	***Não converge***
	PL Modificado	0,48	***Converge***

Tabela 6.26: Tempo de execução utilizando Cenário "B" utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 12s e 240s, respectivamente.

Novamente o algoritmo PL Modificado apresentou um tempo de processamento maior, se comparado aos demais. No entanto, o mesmo é mais estável sem ser muito mais lento, na comparação direta com o Original, que, aliás, somente converge como visto anteriormente quando é empregada a cinemática de movimento proposta no trabalho original [5].

Este tempo de processamento maior para o algoritmo modificado é relativo, uma vez que o algoritmo original apresentava, como já foi dito no item 4.2.4.2, um comportamento divergente, ficando eternamente "preso" no seu *loop* mais interno, sendo necessário forçar sua parada. O tempo de processamento mais baixo no algoritmo

original é devido na realidade à existência do critério de parada ao se chegar ao máximo número de iterações quando o mesmo diverge, e não a alguma melhor característica deste com relação ao modificado.

A tabela 6.27 compila os resultados do tempo médio de processamento em segundos dos algoritmos em estudo quando se diminui o intervalo de tempo de amostragem de 12s para 2s, mantendo-se o tempo total de amostragem em 240s:

Conjunto de Dados da Tabela 6.10 - Cenário "B"	***Algoritmo***	***Tempo Médio em segundos***	***Convergência do Algoritmo***
B1	***EKELUND***	0,33	***Não se aplica***
	PL Original	0,44	***Não converge***
	PL Modificado	3,43	***Converge***
B2	***EKELUND***	0,31	***Não se aplica***
	PL Original	0,45	***Não converge***
	PL Modificado	3,52	***Converge***
B3	***EKELUND***	0,32	***Não se aplica***
	PL Original	0,45	***Não converge***
	PL Modificado	3,32	***Converge***
B4	***EKELUND***	0,30	***Não se aplica***
	PL Original	0,46	***Não converge***
	PL Modificado	4,07	***Converge***
B5	***EKELUND***	0,60	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	2,86	***Converge***

Tabela 6.27: Tempo de execução utilizando Cenário "B" utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

Mais uma vez o algoritmo PL Modificado foi cerca de quase dez vezes mais lento que os demais. No entanto, o PL Original apresenta os mesmos problemas de convergência. O algoritmo modificado mostrou-se estável como já havia ocorrido no passado.

A tabela 6.28 contém a compilação dos resultados do tempo médio de processamento em segundos dos algoritmos para outras corridas com as seguintes condições iniciais: posição inicial do navio observador na origem (0,0); rumos e posições iniciais do contato quaisquer; e velocidades baixas do contato:

Conjunto de Dados da Tabela 6.14 - Cenário "C"	***Algoritmo***	***Tempo Médio em segundos***	***Convergência do Algoritmo***
C1	***EKELUND***	0,50	***Não se aplica***
	PL Original	0,53	***Não converge***
	PL Modificado	5,55	***Converge***
C2	***EKELUND***	0,59	***Não se aplica***
	PL Original	0,49	***Não converge***
	PL Modificado	2,92	***Converge***
C3	***EKELUND***	0,32	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	3,55	***Converge***
C4	***EKELUND***	0,61	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	3,04	***Converge***
C5	***EKELUND***	0,32	***Não se aplica***
	PL Original	0,44	***Não converge***
	PL Modificado	2,98	***Converge***

Tabela 6.28: Tempo de execução utilizando Cenário "C" utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

O fato de ser dez vezes mais lento que os demais não piorou o desempenho do algoritmo PL Modificado, pois o mesmo é estável, fato que não se verificou no original para as corridas que foram criadas. A não convergência deste último é muito pior, pois a estimativa de posição do contato sempre aconteceu de maneira incorreta, como já foi visto nas situações anteriormente estudadas.

A apresentação dos resultados do tempo médio de processamento em segundos dos algoritmos em estudo encontra-se na tabela 6.29, segundo as seguintes condições iniciais: posição inicial do navio observador diferente da origem (0,0); rumos e posições iniciais do contato quaisquer; e velocidades baixas do contato.

Conjunto de Dados da Tabela 6.16 - Cenário "D"	***Algoritmo***	***Tempo Médio em segundos***	***Convergência do Algoritmo***
D1	***EKELUND***	0,62	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	3,18	***Converge***
D2	***EKELUND***	0,33	***Não se aplica***
	PL Original	0,48	***Não converge***
	PL Modificado	4,95	***Converge***
D3	***EKELUND***	0,59	***Não se aplica***
	PL Original	0,46	***Não converge***
	PL Modificado	6,43	***Converge – RMSE aumenta***
D4	***EKELUND***	0,60	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	10,62	***Converge – RMSE aumenta***
D5	***EKELUND***	0,58	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	4,33	***Converge***

Tabela 6.29: Tempo de execução utilizando Cenário "D" utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

Como no caso em estudo o algoritmo PL Modificado apresenta uma deterioração nos seus resultados, principalmente quando o navio observador parte de uma posição muito distante da origem, caso da quarta corrida, somente neste caso o tempo de processamento foi muito alto. Mesmo assim, o algoritmo PL Modificado mostrou-se ainda com robustez e estabilidade, se comparado ao seu parente que não converge como visto em todas as situações estudadas.

Por fim, a tabela 6.30 compila os resultados do tempo médio de processamento em segundos dos algoritmos em estudo, segundo as seguintes condições de inicialização: posição inicial do navio observador na origem (0,0); rumos e posições iniciais do contato quaisquer; e velocidades altas do contato.

Conjunto de Dados da Tabela 6.18 - Cenário "E"	*Algoritmo*	*Tempo Médio em segundos*	*Convergência do Algoritmo*
E1	***EKELUND***	0,61	***Não se aplica***
	PL Original	0,47	***Não converge***
	PL Modificado	3,29	***Converge***
E2	***EKELUND***	0,62	***Não se aplica***
	PL Original	0,42	***Não converge***
	PL Modificado	3,64	***Converge***
E3	***EKELUND***	0,61	***Não se aplica***
	PL Original	0,41	***Não converge***
	PL Modificado	4,05	***Converge – RMSE aumenta***
E4	***EKELUND***	0,61	***Não se aplica***
	PL Original	0,46	***Não converge***
	PL Modificado	4,41	***Converge – RMSE aumenta***
E5	***EKELUND***	0,58	***Não se aplica***
	PL Original	0,44	***Não converge***
	PL Modificado	3,89	***Converge***

Tabela 6.30: Tempo de execução utilizando Cenário "E" utilizando os Algoritmos EKELUND, PL Original e PL Modificado para um intervalo de tempo e tempo total de amostragem de 2s e 240s, respectivamente.

Os resultados encontrados foram mais ou menos iguais aos que já haviam sido encontrados.

Pode-se concluir, portanto que a estabilidade e a robustez são alcançadas em troca de um tempo de processamento um pouco maior, com ordem de grandeza na faixa de oito a dez vezes.

7. Conclusões

Esta Dissertação apresentou um estudo teórico sobre técnicas e algoritmos utilizados em problemas tradicionalmente conhecidos como *Target Motion Analysis* (TMA). Em seguida propôs e desenvolveu um simulador de movimentos de contatos, de modo a fazer a avaliação das técnicas e dos algoritmos inicialmente estudados. Ao final, propôs, implementou e avaliou um novo método que utiliza conceitos e técnicas de inteligência artificial, a fim de estimar e predizer a posição e a velocidade de veículos navais, de superfície e submarinos.

Como em qualquer problema de engenharia, não existe apenas uma solução na resolução do mesmo, mas sim várias soluções, que dependem, *a priori*, dos pressupostos que foram pré-estabelecidos. Portanto, foi necessário estabelecer os mesmos, para que o desenvolvimento do presente trabalho ocorresse de maneira contínua, de modo a conseguir ao seu final, responder as questões indagadas inicialmente.

Os cálculos e as estimativas foram feitos sobre bases de dados, encontradas nos trabalhos pesquisados, que contém valores simulados.

Partindo dos métodos e algoritmos pesquisados na revisão bibliográfica, foi desenvolvido um método híbrido que permite estimar a posição, a velocidade e o rumo de um contato designado, tendo como dados apenas as marcações do contato e as posições do navio observador.

O método proposto utilizou o Estimador Pseudolinear com uma modificação que otimiza o processo de busca pela melhor solução da posição estimada do contato. O Estimador Pseudolinear Modificado (PL-MOD) usou o conceito de guardar o resultado com melhor aptidão e somente alterar o resultado obtido na iteração corrente caso este possua aptidão maior do que a encontrada até então. Este conceito existe nas técnicas de inteligência artificial, como as Técnicas Evolutivas (Algoritmos Genéticos) e Sócio-Cognitivas (*Particle Swarm*).

O processo de análise dos resultados utilizou métricas de desempenho adequadas para realizar a comparação dos resultados alcançados pelas técnicas pesquisadas com os resultados obtidos pela técnica proposta.

Foi demonstrado que a ferramenta desenvolvida é capaz de apresentar desempenho, em termos da precisão da resposta e do tempo para obtenção do resultado, compatível com o estado da arte.

Os resultados obtidos mostraram que o algoritmo PL Original obteve convergência apenas para o padrão de teste utilizado no trabalho no qual foi proposto [5], uma vez que o movimento do navio observador apresentou um *zig-zag* acentuado, o que provoca uma melhor convergência do mesmo. Para os demais casos o mesmo apresenta uma pequena convergência apenas para condições específicas.

Já os resultados alcançados pelo algoritmo PL Modificado foi superior ao Original em todos os demais casos estudados, convergindo de maneira eficiente. O preço pago pela modificação implementada foi um pequeno aumento no tempo de execução da rotina, que, no entanto não causa problemas mesmo no caso de uma aplicação em tempo real.

Um dos resultados alcançados é permitir que as idéias aqui desenvolvidas possam ser empregadas em aplicações reais utilizando a tecnologia disponível e dominada pelo Brasil, uma vez que a total compreensão dos algoritmos de acompanhamento implementados pelos fabricantes dos sistemas operativos em utilização atualmente, no que diz respeito ao seu funcionamento, não é possível apenas com a documentação existente.

Espera-se que este trabalho tenha contribuído com a pesquisa e o desenvolvimento de algoritmos e métodos de acompanhamento de contatos, aperfeiçoando e corrigindo falhas dos modelos até então utilizados, a fim de obter um produto final mais confiável.

Por fim, conclui-se que a contribuição desta Dissertação está em apresentar, ainda que embrionariamente, novas idéias e um novo método PL Modificado, derivado do método PL Original mesclando capacidades de uma e outra técnica, de modo a conseguir gerar novas aplicações (ainda não imaginadas), de modo eficiente.

8. Sugestões para trabalhos futuros

Certamente, uma proposta interessante seria desenvolver uma técnica mista, utilizando outras técnicas de inteligência artificial. No presente trabalho, não foram implementados outros conceitos existentes, mas futuramente poderiam ser implementadas outras variantes das mesmas em partes específicas do algoritmo.

Outra possibilidade seria criar um algoritmo que permitisse detectar quando o contato está executando uma manobra evasiva, como no caso do movimento de *zig-zag*.

Seria possível também implementar o algoritmo DB-TMA, para futuros testes que visam a construção do submarino nuclear brasileiro. Será necessário verificar se o algoritmo DB-TMA apresenta restrições quanto a convergência, de modo semelhante ao que ocorreu com o algoritmo BO-TMA, i.e., o PL Original. Se os problemas quanto à convergência ocorrerem, a solução obtida pelo método PL Modificado poderá ser adaptada para o algoritmo DB-TMA, sendo necessário aplicar diversos padrões de teste, semelhantes aos utilizados nesta dissertação para validar o novo algoritmo DB-TMA.

O algoritmo DB-TMA modificado deve então ser testado novamente para verificar se ficou muito mais lento que a versão atual.

Outra linha de ação possível na busca de algoritmos que trabalhem em problemas de BO seria a de realizar um estudo mais detalhado dos Filtros de Kalman, dos Filtros de Kalman Estendidos - *Extended Kalman Filters* (EKF) e da variante *Unscented Kalman Filters* (UKF). Uma nova modelagem para estes filtros poderia ser proposta para permitir que seja estimada a posição do contato a partir não mais de duas plataformas de observação, mas de apenas uma. Neste caso, o problema da indeterminação talvez pudesse ser resolvido utilizando-se algum artifício de geometria que colocasse uma imagem do navio observador em uma posição na qual seria realizada a medida do complemento do ângulo da marcação atual. Talvez outra solução fosse considerar uma imagem passada das posições ocupadas pelo observador e suas respectivas marcações. Ou ainda alguma implementação híbrida utilizando os outros algoritmos existentes.

Poderia também ser implementado um mecanismo automático de reinicialização do algoritmo PL Modificado, semelhante a um *watchdog* de *hardware*, no caso do mesmo não apresentar a convergência esperada, ou um mecanismo de aceleração de convergência, como a técnica de Aitken e o método de Shanks.

9. Referências Bibliográficas

[1] Pinto, Sílvio Fernando Bernardes. *Um Sistema de Esclarecimento para um Navio Patrulha com Deslocamento de 1300 Toneladas*. Dissertação de Tese de Mestrado em Engenharia Elétrica, Instituto Militar de Engenharia, Rio de Janeiro, 1999.

[2] Ganesh, C., "Fuzzy Logic-based Information Processing in Submarine Combat Systems", In *Fuzzy Information Processing Society de 1999. NAFIPS, 18th International Conference of the North American 10-12 June 1999,* pp. 153 – 157, 1999, USA.

[3] Nguyen, C. T., Ganesh, C. e Gong, K. F., "A Fuzzy Logic-based Intelligent Controller For Contact Management Data Integration", In *Fuzzy Systems Proceedings, 1998. IEEE World Congress on Computational Intelligence, The 1998 IEEE International Conference on Volume 2, 4-9 May 1998*, volume 2, pp. 879 – 884, 1998, USA.

[4] Baldwin, J.F. e Martin, T. P., "Fuzzifying a Target Motion Analysis Model Using Fril and Mathematica", In *Fuzzy Systems Proceedings, 1994. IEEE World Congress on Computational Intelligence, The Third IEEE Conference on 26-29 June 1994,* volume 2 pp.1171 – 1175, 1994, USA.

[5] Ho, K. C. e Chan, Y. T., "An Asymptotically Unbiased Estimator for Bearings-Only and Doppler-Bearing Target Motion Analysis", In *IEEE Transactions on Signal Processing*, volume 54, no. 3, 2006, USA.

[6] Taylor, J. S., Cristianini, N., *Kernel Methods for Pattern Analysis*, 1st Edition, pp. 161-177, Cambrigde University Press, UK, 2004.

[7] Novaes, U. R., *Agrupamento de Dados Através de Algoritmos Swarm*, Tese de Mestrado, COPPE/UFRJ, Rio de Janeiro, 2002.

[8] Savioli, C. E. de F., *Geração Automática de Vetores de Teste Utilizando Técnicas Evolutivas e Sócio-Cognitivas*. Tese de Mestrado, COPPE/UFRJ, Rio de Janeiro, 2005.

[9] Szendrodi, C. E. C., *Síntese de Filtros Analógicos Utilizando Técnicas Evolutivas e Sócio-Cognitivas.* Tese de Mestrado, COPPE/UFRJ, Rio de Janeiro, 2005.

[10] Biscainho, L. W., *Restauração Digital de Sinais de Áudio Provenientes de Gravações Musicais Degradadas.* Tese de Doutorado, COPPE/UFRJ, Rio de Janeiro, 2001.

[11] Burrus, C. S., Gopinath, R. A., Guo, H., & Al., Editor: Primer, A., *Introduction to Wavelets and Wavelet Transforms*., Prentice Hall, Upper Saddle River, New Jersey, 1996.

[12] Diniz, P. S. R.., da Silva, E. A. B., Lima Netto, S., *Processamento Digital de Sinais: – Projeto e Análise de Sistemas*., Artmed Editora S.A. – Bookman Companhia Editora, Porto Alegre, Brasil, 2004.

[13] Chen, W. K. & Al., *Circuits & Filters Handbook*, 2nd Edition, CRC Press, USA, 2003.

[14] Mallat, S., *A Wavelet Tour of Signal Processing*, 2nd Edition, Academic Press, USA, 1999.

[15] Daubechies, I., *Ten Lectures on Wavelets*, SIAM, Philadelphia, PA, USA, 1992.

[16] Thuillard, M., *Wavelets in Soft Computing, World Scientific Series in Robotics and Intelligent Systems,* Vol. 25, pp. 28-29 World Scientific Publishing Co. Pte. Ltd., USA, 2001.

[17] Koza, John R., *Genetic Programming: on the programming of computers by means of natural selection*. The MIT Press, 5th printing, 1996.

[18] Zebulum, Ricardo S., Pacheco, M.A.C. and Vellasco, M.M.B.R.. *Evolutionary Electronics: Automatic Design of Electronic Circuits and Systems by Genetic Algorithms*. CRC Press, 2002.

[19] Kennedy, J., Eberhart, R. C., *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.

[20] Eberhart, R. C., Kennedy, J., *A New Optimizer using Particle Swarm Theory*, Proceedings of the 6th International Symposium on Micro-Machine and Human Science, pp39-43, Nagoya, Japan, 1995.

[21] Holland, J. H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Harbor, 1975.

[22] Haykin, S., *Redes Neurais: princípios e prática*, 2ª. edição, Bookman, Porto Alegre, 2001.

[23] Larson, H. J., *Introduction to Probability Theory and Statistical Inference*, John Wiley and Sons, 1982.

[24] Hanselman, D., Littlefield, B. *MATLAB 5 Versão do Estudante: Guia do Usuário*. Makron Books do Brasil, 1999.

[25] Kalman, R. E., *A new approach to linear filtering and prediction problems*. In *Transactions of the ASME*, *Journal of Basic Engineering*, no. 82, série D, pp.34 – 45, 1960, USA.

[26] Särkkä, S., *Recursive Bayesian Inference On Stochastic Differential Equations*., D.Sc. dissertation, Helsinki University of Technology, FINLAND, 2006.

[27] Haykin, S., *Kalman Filtering and Neural Networks*, 1st Edition, John Wiley & Sons, Inc., USA, 2001.

[28] Grewal, M. S., Andrews, A. P., *Kalman Filtering: Theory and Practice – Using MATLAB*, 2nd Edition, John Wiley & Sons, Inc., USA, 2001.

[29] Oliveira, J. R. P. de, *Acompanhamento de Alvos-Radar Utilizando Filtragem de Kalman e Vetor de Estados com Dimensão Variável.* Tese de Mestrado, COPPE/UFRJ, Rio de Janeiro, 2005.

[30] Página Oficial da Marinha do Brasil - http://www.mar.mil.br/ - link: Amazônia Azul.

[31] Página Oficial da Petrobrás - http://www.petrobras.com.br/ - link: Espaço Conhecer.

[32] Página Comercial de Macaé - http://www.clickmacae.com.br/home.asp - link: Bacia de Campos.

[33] Maia, Viviane, *Imensidão azul – A Amazônia marítima brasileira*, Crea-RJ em Revista, número 61, pp. 14 -15, julho de 2007, Rio de Janeiro.

[34] Pedrosa, José Julio, Editorial: *Levantamento da Plataforma Continental*, Revista do Clube Naval, número 343, pp. 4 -5, JUL/AGO/SET 2007, Rio de Janeiro.

[35] Seção Datas, Revista VEJA, pp. 100, Edição 2020, número 31, 8 de agosto de 2007, São Paulo. – página: http://veja.abril.com.br/080807/datas.shtml

[36] Página Oficial do Ministério do Desenvolvimento, Industria e Comércio Exterior- http://www.desenvolvimento.gov.br/portalmdic/sitio/ - link: Comércio Exterior - Estatísticas de comércio exterior - Balança comercial - dados consolidados.

[37] Página: *Wavelets e as Artes Multiresolucionárias.* Laboratório de Sistemas Integráveis (LSI) do Depto. de Engenharia Elétrica da Escola Politécnica da USP - Prof. Regis Rossi A. Faria - http://www.lsi.usp.br/~regis/wlets.html.

[38] de Lima, P. C., *Wavelets em Processamento de Imagens*, Departamento de Matemática – ICEX – UFMG, 2002.

[39] *Help - MATLAB* 7.1.0.246 (R14) - *Service Pack 3 for Windows XP,* The MathWorks, Inc, 2005.

[40] Misiti, M., Misiti, Y., Oppenheim, G., Poggi, J. M., *Wavelets Toolbox for Use with MATLAB – User´s Guide*, 2nd Version, The MathWorks, Inc, 2002.

APÊNDICE I. ARTIGOS

Neste apêndice são apresentados alguns artigos sobre o assunto desta tese que foram apresentados em congressos e workshops, bem como artigos de submissão já aceita.