

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**POSICIONAMENTO EM AMBIENTES NÃO  
ESTRUTURADOS E TREINAMENTO DE REDES  
NEURAS UTILIZANDO FILTROS DE KALMAN**

**DENIS PEREIRA DE LIMA**

**ORIENTADOR: PROF. DR. EDILSON REIS RODRIGUES KATO**

São Carlos - SP

Março/2016

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**POSICIONAMENTO EM AMBIENTES NÃO  
ESTRUTURADOS E TREINAMENTO DE REDES  
NEURAS UTILIZANDO FILTROS DE KALMAN**

**DENIS PEREIRA DE LIMA**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Edilson Reis Rodrigues Kato

São Carlos - SP

Março/2016

Ficha catalográfica elaborada pelo DePT da Biblioteca Comunitária UFSCar  
Processamento Técnico  
com os dados fornecidos pelo(a) autor(a)

L732p Lima, Denis Pereira de  
Posicionamento em ambientes não estruturados e  
treinamento de redes neurais utilizando filtros de  
Kalman / Denis Pereira de Lima. -- São Carlos :  
UFSCar, 2016.  
150 p.

Dissertação (Mestrado) -- Universidade Federal de  
São Carlos, 2016.

1. Filtros de Kalman adaptativos. 2. Séries  
caóticas. 3. Unscented Kalman Filter. 4. Redes  
neurais. 5. Filtro de Kalman estendido. I. Título.



UNIVERSIDADE FEDERAL DE SÃO CARLOS

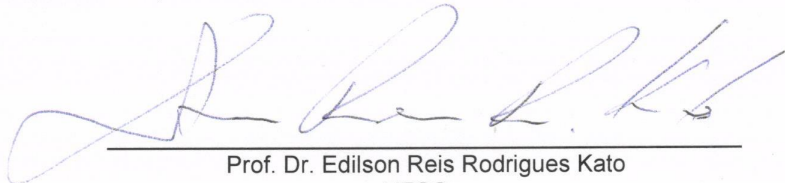
Centro de Ciências Exatas e de Tecnologia  
Programa de Pós-Graduação em Ciência da Computação

---

**Folha de Aprovação**

---

Assinaturas dos membros da comissão examinadora que avaliou e aprovou a Defesa de Dissertação de Mestrado do candidato Denis Pereira de Lima, realizada em 04/03/2016:



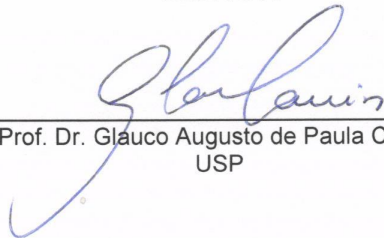
---

Prof. Dr. Edilson Reis Rodrigues Kato  
UFSCar



---

Prof. Dr. Ricardo Yassushi Inamasu  
EMBRAPA



---

Prof. Dr. Glauco Augusto de Paula Caurin  
USP

*alea jacta est*

*Julius Caesar*

# AGRADECIMENTOS

Agradeço a Deus por permitir a conclusão de mais uma etapa em minha vida.

Agradeço a minha família e minha esposa pelo apoio e suporte necessários para conclusão desse trabalho.

Agradeço ao Prof. Dr. Edilson R. R. Kato meu orientador pela confiança, apoio e com conhecimento para o desenvolvimento desse trabalho.

Agradeço ao Prof. Dr. Emerson C. Pedrino, seus alunos Manoel Aranda de Almeida, Paulo Paris e todos os envolvidos que auxiliaram no desenvolvimento de trabalhos realizados durante essa pesquisa e que conseqüentemente também contribuíram com essa pesquisa.

Agradeço a meus amigos Cedrick Bamba, Amir Jallifard, Rafael Sanches e todos os colegas de laboratório.

Agradeço ao Departamento de Física da UFSCar, na pessoa do Prof. Dr. Celso Villas Boas Chefe do Departamento, pela disponibilidade de horários para desenvolvimento dessa pesquisa.

Agradeço ao Prof. Dr. Estevam Hruska, Prof. Dr. Adenilson José Chiquito, Prof. Dr. Adalberto Picinin e todos os professores do Departamento de Física e Computação da UFSCar que contribuíram para o desenvolvimento desse trabalho.

# RESUMO

Filtros de Kalman estão consagrados na literatura técnica, como uma das formas de prever novos estados em sistemas não-lineares, fornecendo uma solução recursiva para o problema da filtragem ideal linear. Após 56 anos de sua descoberta, muitas modificações e melhorias foram propostas, procurando obter uma maior precisão e velocidade na predição de novos estados. Algumas dessas mudanças são utilizadas neste trabalho; sendo elas o Filtro de Kalman Estendido (EKF), Unscented Kalman Filter (UKF) e Filtro de Kalman de Cubagem Esférica Radial (CKF). O objetivo deste trabalho, dividido em três partes distintas, porém complementares: Implementação/Análise comparativa da predição dos Filtros de Kalman em sistemas complexos (Series), Análise qualitativa das possíveis utilizações das variantes do Filtro de Kalman para treinamento de Redes Neurais e Determinação de posição e velocidade de um objeto deslocado sobre um plano simulado. Possuindo essas análises papel fundamental na fomentação dos estudos citados na literatura científica durante o trabalho, e comprovando a possibilidade desses algoritmos/métodos serem utilizados em tarefas de posicionamento em ambientes não estruturados.

**Palavras-chave:** *Filtros de Kalman Adaptativos; Séries Caóticas; Unscented Kalman Filter; Redes Neurais; Filtro de Kalman Estendido; Predição de Estado, Ambientes não estruturados.*

# ABSTRACT

Kalman filters are rooted in the technical literature, as a way of predicting new states in nonlinear systems providing a recursive solution to the problem of linear optimal filtering. Therefore, 56 years after its discovery, many modifications have been proposed in order to obtain better accuracy and speed. Some of these changes are used in this work; these being the Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF) and Kalman Filter Cubature (CKF). This work, divided into three distinct parts: Implementation / Comparative analysis of prediction of Kalman filters in complex systems (Series), qualitative analysis of the possible uses of the Kalman filter variants for neural network training and position and velocity determination a displaced object on a simulated plane with some trajectories Having these analyzes key role in fostering the studies cited in the scientific literature, proving the possibility of such algorithms and methods are used for positioning in unstructured environments.

**Keywords:** *Adaptive Kalman Filters; Chaotic Series; Unscented Kalman Filter; Neural Networks; Extended Kalman Filter; State Estimation, Unstructured Environment.*



# LISTA DE FIGURAS

Figura 2-1 - Operação do Filtro de Kalman .....	24
Figura 2-2 - Operação do Filtro Estendido .....	27
Figura 2-3 Representação dos Filtros EKF e UKF. ....	31
Figura 2-4 Funções de Transferência. (ZUBEN, 2001) .....	40
Figura 3-1 - Erro Médio Quadrático da Posição. ....	49
Figura 3-2 - Erro Médio Quadrático da Velocidade. ....	49
Figura 5-1 - Estrutura de uma Rede Neural Multicamadas. ....	69
Figura 6-1 - Unscented Kalman Filter de Segunda Ordem. ....	81
Figura 6-2 - Extended Kalman Filter.....	81
Figura 6-3 - Unscented Kalman Filter de Segunda Ordem. ....	82
Figura 6-4 - Unscented Kalman Filter de Primeira Ordem. ....	82
Figura 6-5 - Cubature Kalman Filter. ....	83
Figura 6-6 – Treinamento de RN utilizando o EKF e UKF ( $\alpha$ and $\beta = 10^{-2}$ ). ....	86
Figura 6-7 -Treinamento de RN utilizando o EKF e UKF ( $\alpha$ and $\beta = 10^{-3}$ ). ....	88
Figura 6-8 - Trajeto com curvas e sensores. ....	91
Figura 6-9 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.1).....	93
Figura 6-10 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.1).....	93
Figura 6-11 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.1).....	94
Figura 6-12 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.2).....	97
Figura 6-13 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.2).....	97
Figura 6-14 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.2).....	98
Figura 6-15 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.3).....	100
Figura 6-16 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.3).....	100
Figura 6-17 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.3).....	101

Figura 6-18 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.4).....	103
Figura 6-19 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.4).....	104
Figura 6-20 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.4).....	105
Figura 6-21 - Comparação entre a Posição dos Sensores x Erros (Trajetória 1)....	106
Figura 6-22 - Trajeto Circular .....	107
Figura 6-23 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.1).....	108
Figura 6-24 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.1).....	109
Figura 6-25 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.1).....	110
Figura 6-26 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.2).....	112
Figura 6-27 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.2).....	113
Figura 6-28 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.2).....	113
Figura 6-29 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.3).....	116
Figura 6-30 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.3).....	116
Figura 6-31 -Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.3).....	117
Figura 6-32 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.4).....	119
Figura 6-33 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.4).....	120
Figura 6-34 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.4).....	121
Figura 6-35- Comparação entre a Posição dos Sensores x Erros (Trajetória 2)....	122
Figura 6-36 – Trajeto com mudanças bruscas de direção. ....	123
Figura 6-37 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.1).....	124
Figura 6-38 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.1).....	125
Figura 6-39 - Posicionamento utilizando o UKF e URTS (Trajetória 3 e Pos.1).....	126

Figura 6-40 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.2).....	127
Figura 6-41 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.2).....	128
Figura 6-42 - Posicionamento utilizando o UKF e URTS (Trajetória 3 e Pos.2).....	128
Figura 6-43 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.3).....	130
Figura 6-44 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.3).....	131
Figura 6-45 - Posicionamento utilizando o UKF e UKS (Trajetória 3 e Pos.3). ....	132
Figura 6-46 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.4).....	134
Figura 6-47 - Posicionamento utilizando o EKF e EKS de 2º ordem (Trajetória 3 e Pos.4).....	135
Figura 6-48 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.4).....	135
Figura 6-49 - Comparação entre a Posição dos Sensores x Erros (Trajetória 3). ...	136

# LISTA DE TABELAS

Tabela 2-1 - Filtro de Kalman Discreto atualização temporal.....	23
Tabela 2-2 - Filtro de Kalman Discreto atualização das medições.....	23
Tabela 3-1 - Comparação de diferentes modificações (ZHAN e WAN, 2006).....	47
Tabela 3-2 - Trabalhos relevantes realizados utilizando filtros de Kalman.....	59
Tabela 6-1 Comparação sobre as diferentes modificações do Filtro de Kalman.....	84
Tabela 6-2 - Treinamento / Número de Neurônios .....	85
Tabela 6-3 Treinamento / Número de Neurônios 2 .....	87
Tabela 6-4 - Erros Posicionamento (Trajetória 1 e Pos.1) .....	95
Tabela 6-5 - Erros Posicionamento (Trajetória 1 e Pos.2) .....	98
Tabela 6-6 - Erros Posicionamento (Trajetória 1 e Pos.3) .....	102
Tabela 6-7 - Erros Posicionamento (Trajetória 1 e Pos.4) .....	105
Tabela 6-8 - Erros Posicionamento (Trajetória 2 e Pos.1) .....	111
Tabela 6-9 - Erros Posicionamento (Trajetória 2 e Pos.2) .....	114
Tabela 6-10 - Erros Posicionamento (Trajetória 2 e Pos.3) .....	118
Tabela 6-11 - Erros Posicionamento (Trajetória 2 e Pos.4) .....	121
Tabela 6-12 - Erros Posicionamento (Trajetória 3 e Pos.1) .....	126
Tabela 6-13 - Erros Posicionamento (Trajetória 3 e Pos.2) .....	129
Tabela 6-14 - Erros Posicionamento (Trajetória 3 e Pos.3) .....	132
Tabela 6-15 - Erros Posicionamento (Trajetória 3 e Pos.4) .....	136

# LISTA DE ABREVIATURAS E SIGLAS

**2D** – Duas Dimensões.

**3D** – Três Dimensões.

**CAD** – Computer Aided Design (Projeto Auxiliado por Computador)

**CAM** – Computer Aided Manufacture (Manufatura Auxiliada por Computador)

**CKF** – *Cubature Kalman Filter*

**EKF** – Filtro de Kalman Estendido

**GHKF** – *Gauss Hermite Kalman Filter.*

**GPS** – Global Positioning System (Sistema de Posicionamento Global)

**IA** – Inteligência Artificial

**KF** – *Kalman Filter* (Filtro de Kalman)

**MSE** – Mean Square Error (Erro Médio Quadrático)

**RMSE** – Root Mean Square Error (Desvio Padrão)

**RN** – Rede Neural.

**UKF** – *Unscented Kalman Filter.*

**GRV** -*Gaussian Random Variable*

# SUMÁRIO

<b>INTRODUÇÃO .....</b>	<b>13</b>
1.1 Motivação.....	18
1.2 Objetivos Gerais.....	20
1.3 Objetivos Específicos.....	20
1.4 Organização do Trabalho.....	21
<b>FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>22</b>
2.1 Filtro de Kalman .....	22
2.2 Filtro de Kalman Estendido (EKF).....	25
2.3 Filtro de Kalman Unscented (UKF).....	27
2.4 Suavizador Rauch Tung Striebel.....	32
2.5 Redes Neurais.....	33
2.5.1 Conceitos .....	35
2.5.2 Importância das Redes Neurais .....	35
2.5.3 Funções de Ativação .....	38
2.5.4 Treinamento e Arquiteturas de RN's .....	41
2.6 Bearing Only.....	43
<b>REVISÃO BIBLIOGRÁFICA.....</b>	<b>45</b>
3.1 Revisão Bibliográfica .....	45
3.2 Trabalho Correlacionados .....	50
3.3 A Survey of Object Recognition Methods for Automatic Asset Detection in High-Definition Video (WARSOP, 2010).....	51

3.4 Road sides recognition in non-structured environments by vision (AUFRKRE, AUFRKRE, <i>et al.</i> , 2004). .....	52
3.5 Movement-Flow-Based Visual Servoing and Force Control Fusion for Manipulation Tasks in Unstructured Environments (TORRES, 2005). .....	53
3.6 Simple, Robust Autonomous Grasping in Unstructured Environments (DOLLAR e HOWE, 2007). .....	54
3.7 Probabilistic Scan Matching for Motion Estimation in Unstructured environments (MONTESANO, MINGUEZ e MONTANO, 2005). .....	55
3.8 Appearance-Based Visual Learning in a Neuro-Fuzzy Model for Fine-Positioning of Manipulators (ZHANG, SCHMIDT e KNOLL, 1999). .....	56
3.9 Distributed Multiple View Fusion for Two-Arm Distance Estimation (SCHEERING, ZHANG e KNOLL, 1999). .....	56
3.10 Targeted On-line Modeling for an Extended Kalman Filter Using Artificial Neural Networks (STUBBERUD e OWEN, 1998). .....	57
3.11 Filtros de Kalman e suas aplicações. ....	58
3.12 Considerações Finais .....	60
<b>PROPOSTA DO TRABALHO .....</b>	<b>61</b>
4.1 Redes Neurais.....	62
4.2 Análise de Performance do Sistema Neural e Posicionamento .....	63
4.3 Testes de Posicionamento em Simulador .....	63
4.4 Resultados Esperados .....	64
4.5 Delimitação do Escopo.....	65
<b>IMPLEMENTAÇÃO .....</b>	<b>66</b>
5.1 Algoritmo do Filtro EKF .....	66
5.2 Treinamento utilizando o filtro EKF .....	68

5.3 Algoritmo do Filtro UKF .....	71
5.4 Treinamento utilizando o Filtro UKF .....	73
5.5 Modelo de Posicionamento usando sensores angulares (Bearing Only) .....	75
<b>RESULTADOS OBTIDOS .....</b>	<b>78</b>
6.1 Avaliação das Modificações do Filtro de Kalman .....	78
6.2 Resultados Treinamento das RN's .....	84
6.3 Resultados Estruturação Ambiental (Bearing Only) .....	89
6.3.1 Resultados (Deslocamento linear com curvas) .....	91
6.3.2 Resultados (Deslocamento Circular) .....	107
6.3.3 Resultados (Deslocamento com Mudanças Bruscas de Direção) .....	123
6.4 Considerações Finais .....	137
<b>CONCLUSÃO .....</b>	<b>138</b>
7.1 Trabalhos Futuros .....	142
<b>REFERÊNCIAS.....</b>	<b>143</b>



# Capítulo 1

## INTRODUÇÃO

---

---

A partir da Revolução Industrial, ocorrida na Inglaterra no século XVIII, houve grande demanda por sistemas de manufatura, com maior capacidade de produção e qualidade, isto aliado a redução de tempo de produção e segurança durante o processo.

Na resolução dos problemas referentes a tempo e segurança em sistemas de manufatura, destacam-se a automação programável, relativa à fabricação em série de pequenos e médios lotes de produtos, e a automação flexível, referida à fabricação de lotes variáveis de produtos diversos (ROMANO e FERREIRA, 2002).

Segundo Koren (1985), a grande revolução que ocorreria nas técnicas de produção e gestão até o final do século vinte, seria uma fábrica do futuro onde desde a concepção do produto até a fabricação, montagem e inspeção de produtos, seria monitorado e controlado por computadores e realizada por robôs industriais e sistemas inteligentes, se tornando em parte realidade nos dias atuais.

Num contexto industrial pode se definir a automação como a tecnologia que se ocupa da utilização de sistemas mecânicos, eletroeletrônicos e computacionais na operação e controle da produção (PAZOS, 2002).

Exemplos dessa tecnologia de automação incluem máquinas de montagem mecanizadas, sistema de controle de realimentação, máquinas ferramentas dotadas de controle numérico e robôs (GROOVER, 2010).

Os robôs industriais têm sido utilizados nos processos de automação programável e flexível, pois são essencialmente máquinas capazes de realizar os mais diversos movimentos programados (ROMANO e FERREIRA, 2002).

No ambiente industrial o uso de robôs se tornou frequente, estes passaram a ser identificados como um dispositivo de manufatura na década de 1960, juntamente com o projeto auxiliado por computador (CAD) e sistemas de manufatura auxiliada por computador (CAM), caracterizou as últimas tendências na automação da manufatura de processos, essas tecnologias estão levando automação industrial a um novo patamar, cujo alcance é desconhecido (CRAIG, 2005).

A utilização dos robôs ainda é muito limitada, motivada principalmente, pela incapacidade técnica dos mesmos em realizar tarefas em ambientes heterogêneos e não estruturados, sendo esses definidos por Vatavu e Nedevschi (2013) e Nemec e Zlajpah (2002), como sendo ambientes com medidas incorretas, oclusões, objetos deformados, comportamento imprevisível, obstáculos não conhecidos com antecedência ou que mudam de posição ao longo do tempo.

Quando se trabalha com a extração de medidas do ambiente a partir de imagens é necessário ter em consideração que as estruturas geométricas devem ser confiáveis e bem definidas do cenário a ser visualizado (WARSOP, 2010).

As estruturas geométricas podem não estar bem definidas em uma ou várias imagens devido a problemas no ajuste do foco ou por fatores mais complexos como o *Motion Blur*, observado em imagens obtidas com um tempo de exposição longo ou por objetos que se movem rapidamente em relação à câmera. A problemática da oclusão também interfere na visibilidade de estruturas geométricas confiáveis, sendo esse fator mais crítico na utilização de múltiplas vistas de câmara, pois, dificulta o tratamento da correspondência (SILVA, 2010).

Existem diversos problemas na estimação de posição em ambientes não estruturados supervisionados por visão computacional, tais como estruturação ambiental em 2D, a falta de parâmetros relacionados com a profundidade dos objetos a serem classificados e qualificados os quais trazem dificuldades ao desenvolvedor, segundo Weckesse e Dillmann (1998) e Janabi-Sharifi e Marey (2010), uma das

soluções para este problema é a utilização de correspondências 2D e 3D para a obtenção das geometrias do ambiente estudado.

O ambiente pode ser estruturado e ser suprimida a falta da variável profundidade utilizando sensores complementares a visão computacional como por exemplo: Sensores Ultrassom, Lasers, GPS, Sensores Capacitivos, Sensores Indutivos, Transdutores de Torque, Impedância e Corrente Elétrica (CHEN, 2012).

Os Filtros de Kalman tornaram-se uma abordagem padrão na redução dos erros intrínsecos associados em sistemas de visão robótica e no uso em medições de variáveis advindas de diferentes fontes. Os problemas gerados por erros, ruídos e dados incompletos captados por câmeras sem calibração tem como solução a utilização de uma regra crucial para o sucesso, a metodologia conhecida como Reconhecimento de Padrões (CHEN, 2012) (JANABI-SHARIFI e MAREY, 2010).

Este filtro se integra de forma simples em sistemas híbridos, que utilizam observações de recursos visuais com outras medições de GPS, sensores inerciais ou sensores de força. O filtro calcula com precisão a posição relativa e velocidade, utilizando uma forma adaptativa de recursos e em tempo real (CHEN, 2012).

O desenvolvimento do Filtro de Kalman tem início em 1949, quando Norbert Wiener revolucionário da teoria de probabilidades, tornou-se um dos nomes mais famosos que promoveram a cibernética. Seu grande avanço na matemática, foi a extensão da Transformada de Fourier de Energia Finita em função, sendo esta a primeira teoria de estimação ideal para sistemas que envolvem processos randômicos (GREWAL e ANDREWS, 1993) (WIENER, 1949).

Porém o Filtro de Wiener, não obtém bons resultados em sistemas com não linearidades, desse modo Rudolph Emil Kalman em 1960, derivou a solução para o problema de filtragem de Wiener, utilizando o modelo de espaço de estados para os processos dinâmicos e aleatórios. Conseguindo soluções mais eficientes em sistemas de navegação e de previsão de novos estados, especialmente em aplicações de sistemas de navegação autônoma e na previsão de séries temporais (KALMAN, 1960) (BISHOP, 2006).

Teoricamente ele é um estimador, inicialmente chamado “*The Linear-Quadratic-Gaussian Problem*”, com o propósito de estimar instantaneamente o estado de um sistema linear dinâmico perturbado por um ruído branco gaussiano, sendo ele certamente uma das grandes descobertas na Teoria da Estatística de Previsão (GREWAL e ANDREWS, 1993).

O cerne do Filtro de Kalman é um algoritmo que utiliza uma série de medições observadas ao longo do tempo, com ruído (flutuações aleatórias geradas pelo sistema), outras imprecisões e falta de dados, para a produção de estimativas de variáveis desconhecidas que tendem a gerar informações mais precisas, com base em uma única medida (BISHOP, 2006) (HAYKIN, 2001) (GREWAL e ANDREWS, 1993).

Esses filtros se firmaram na formulação do espaço de estados em sistemas dinâmicos lineares, fornecendo uma solução recursiva para o problema de filtragem ideal linear, podendo ser utilizados em sistemas estacionários e não-estacionários, o estado é calculado computando o anterior e a nova entrada, somente o estado anterior necessita ser armazenado, como todos os dados passados não necessitam serem armazenados a eficiência computacional aumenta, ao invés de se estimar o novo estado utilizando todos os dados anteriormente observados, a cada novo passo do processo de filtragem (HAYKIN, 2001).

Muitos estudos têm sido dedicados à melhoria do Filtro de Kalman, ou seja, a melhoria da estabilidade numérica, a redução do tempo de computação ou o estudo de implementações efetivas no aumento do desempenho (AUGER, HILAIRET, *et al.*, 2013).

Alterações no Filtro de Kalman, as quais se referem a estimação da matriz de covariância Ramakoti, Vinay E Jatoth (2009), foram noticiadas em 51 anos de existência dos Filtros de Kalman, onde foram relatadas mais de 20 variantes para aplicações relacionadas a visão robótica (CHEN, 2012).

O Filtro de Kalman aborda o problema geral de tentar estimar o estado de um processo discreto, que é governado por uma Equação Diferencial Estocástica. Mas o que pode ser feito se o processo a ser controlado tem características não lineares, há algum tempo atrás, depois de uma intensa pesquisa foi construído uma das mais bem-

sucedidas alterações do filtro de Kalman, o filtro que lineariza sobre a média e covariância que agora é conhecido como o Filtro de Kalman Estendido (EKF) (GREWAL e ANDREWS, 1993) (HAYKIN, 2001) (MAYBECK, 1982).

Assim, o EKF assume que há um ruído branco gaussiano, no entanto, se houver ruídos coloridos ou erros sistemáticos no sistema de posicionamento, câmeras ou outros sistemas não lineares, o EKF pode não garantir a precisão da estrutura e detecção de movimento. Se os erros associados a sensores são desconhecidos, com a atualização temporal do filtro pode-se gerar erros que tendem a aumentar inesperadamente (JULIER e UHLMANN, 1997).

Em resposta a esses erros de modelagem e sistemáticos, surgem novas modificações dos Filtros de Kalman, nomeados Unscented Kalman Filter (UKF) e Cubature Kalman Filter (CKF) (CHEN, 2012).

A operação vital realizada no UKF proposto por Julier e Uhlmann é a propagação de uma variável aleatória gaussiana através da dinâmica do sistema, utilizando uma abordagem de amostragem determinística (WAN e VAN DER MERWE, 2000).

Utilizando o princípio de que um conjunto de pontos discretamente amostrados podem ser usadas para parametrizar o erro e a covariância, o rendimento desse estimador tem desempenho equivalente ao Filtro de Kalman para sistemas lineares, e ainda generaliza de forma eficiente para sistemas não-lineares, sem os passos de linearização exigidos pelo EKF (JULIER e UHLMANN, 1997).

O Filtro CKF proposto por Arasaratnam e Simon (2009), tem seu cerne baseado na regra de Cubagem Esférico Radial, o que torna possível calcular numericamente integrais multivariadas, encontradas no filtro Bayesiano não-linear, fornecendo um conjunto de pontos de Cubagem escalados linearmente com a dimensão do estado vetor.

O uso dos Filtros de Kalman e suas várias modificações trouxeram grandes avanços nas áreas da Robótica e Cibernética. O estudo da previsão de novos estados em sistemas não-lineares tem grande utilidade em sistemas de posicionamento em ambientes não estruturados (CHEN, 2012).

A adição de Redes Neurais (RN) ao Filtro de Kalman e suas modificações, vem sendo objeto de estudo por diversos pesquisadores Arasaratnam e Haykin( 2008), Haykin (2001), Jeong-Gwan Kang (2010), Stubberud e Owen (1998),Stubberud, Lobbia e Owen (1995), Zhan e Wan (2006),Zhong, Zhong e Peng (2013), tentando obter um melhor desempenho do filtro de Kalman para sistemas com características caóticas ou potencialmente caóticas, essas pesquisas resultaram em modelos com alta precisão, velocidade e acurácia. A comparação entre esses modelos se tornou um ponto de referência para determinar a sua eficácia e desempenho.

Desta forma esta proposta aborda de forma sistemática as características recorrentes em series potencialmente caóticas e sistemas de servoposicionamento e/ou servovisão, as não linearidades inseridas seja por dados incompletos, ruídos ou erros aleatórios não previsíveis, tendo como objetivo determinar a melhor modificação do Filtro de Kalman a ser utilizada para a Estruturação Ambiental e Treinamento de Redes Neurais.

## **1.1 Motivação**

De certo modo a apresentação inicial dos problemas relacionados a servovisão e a utilização Filtro de Kalman e suas principais modificações para solucionar ou mitigar essas falhas, definem o escopo inicial da pesquisa.

A grande utilização dos Filtros de Kalman ao longo de 51 anos e aproximadamente 800 publicações sobre na área de Servovisão (CHEN, 2012), demonstram o seu efetivo uso em diversas áreas do conhecimento e a propensão a novas abordagens a serem estudadas.

O uso do filtro de Kalman em sistemas de servovisão com a adição das ferramentas da área de inteligência artificial como Algoritmos Genéticos, Redes Neurais, Lógica Nebulosa e Algoritmos Bioinspirados, trazem maior eficácia e precisão na localização de objetos, posicionamento robótico, filtros de imagem ou definição de trajetórias (CHEN, 2012).

Os trabalhos propostos por Zhan e Wan (2006), Haykin (2001), Bar-Shalom, Li e Kirubarajan (2001), Corke e Good (1993), Särkkä (2007) e Chen (2012), trazem uma nova perspectiva para proposição de algoritmos eficientes nessas tarefas.

Baseando-se que em sistemas de visão existe grande propensão a perda de dados e ruídos durante sua aquisição e processamento, sendo também demonstrado que o Filtro de Kalman em sua forma básica, possui limitação quanto a predição ótima em sistemas não lineares, devido a sua ineficiência na linearização de equações diferenciais, a proposta de utilizar metodologias da Inteligência Artificial (Redes Neurais, Algoritmos Bioinspirados, Fuzzy, etc.) e/ou modificações com maior efetividade nas linearizações (EKF, UKF e CKF) , para mitigar estes problemas e melhorar a percepção e localização de objetos em sistemas de Servovisão vem sendo objeto de intensa pesquisa ao longo dos anos (CHEN, 2012).

A base racional para a combinação das modificações do Filtro de Kalman e Redes Neurais em um sistema híbrido, é o fato de possuírem características complementares, no que diz respeito à previsão de novos estados em séries temporais ou sistemas de posicionamento, com a presença de ruído e considerando características não-lineares, potencialmente caóticas, aleatórias ou estocásticas.

O método de análise estatística do Filtro de Kalman pode minimizar a influência do ruído, pois este trabalha com a variância do ruído nos dados extraídos do sistema ruidoso. Este cálculo da variância é utilizado para melhorar o prognóstico, juntamente com a matriz de covariância do erro de predição.

A razão para utilizar o EKF, UKF ou CKF esta no tratamento de não-linearidades no cálculo numérico do modelo e a possibilidade de interagir com a RN na geração da função que descreva o sistema ou serie estudada.

O principal problema dos métodos ou filtros estatísticos, como EKF, UKF ou CKF é a dificuldade de criar uma abordagem complexa da função geradora, pela falta de compreensão de certos modelos reais, onde muitas características e parâmetros não são conhecidos. Podendo estes problemas serem suprimidos pelo uso de uma RN assistente no que diz respeito a função geradora da série ou sistema.

Sendo utilizado para posicionamento sensores (Ultrassom, Laser e Doppler) em sistemas de posicionamento em ambientes aquáticos, escuros e de difícil estruturação ambiental.

Sendo este um dos principais motivos para determinação de uma metodologia eficiente na localização de objetos nesses ambientes de difícil percepção espacial.

## **1.2 Objetivos Gerais**

O objetivo desse trabalho é o estudo de uma metodologia que possibilite o posicionamento robótico em ambientes não estruturados, isto é, ambientes com características que o tornem propenso a ruídos ou erros durante os processos de amostragem e/ou posicionamento.

Nossa pesquisa objetiva avaliar a possibilidade na utilização de Filtros de Kalman e suas modificações, em paralelo ou de forma corretiva em Sistemas de Posicionamento utilizando Sensores de Rolamento (Angulares), tentando determinar a velocidade e trajetória do objeto deslocando-se sobre o plano de trabalho.

Tentando desse modo obter uma abordagem que resolva com maior acuracidade e precisão os problemas abordados em sistemas de posicionamento em ambientes não estruturado.

## **1.3 Objetivos Específicos.**

- Determinação da Modificação do Filtro de Kalman com maior precisão e acuracidade na predição de Séries.
- Avaliação Qualitativa das Modificações do Filtro.
- Treinamento das Redes Neurais utilizando filtros de Kalman.
- Avaliação Qualitativa do Treinamento de Redes Neurais.
- Simulação em Ambiente Controlado de Posicionamento.



- Utilização da Metodologia *Bearing Only*.
- Determinar a resposta do método em relação a três tipos de movimentos simulados e posição espacial dos sensores, obtendo velocidade e posição de um objeto.

## 1.4 Organização do Trabalho

- No Capítulo 2, é apresentado uma revisão teórica inicial com relevância sobre o assunto abordado, trazendo ao leitor embasamento teórico ao entendimento da pesquisa.
- No Capítulo 3, trabalhos relevantes efetuados na área de Servovisão e Modificações do Filtro de Kalman, baseados em diversas bibliografias, exibindo a viabilidade do objeto em estudo.
- O Capítulo 4 apresenta a proposta para a presente pesquisa, para demonstrar os passos até variante do algoritmo em evidência, são expostos e exemplificados a fim de demonstrar o caminho a ser seguido durante a pesquisa proposta. Os detalhes sobre as metodologias de IA a serem utilizadas no processo de construção da variante dos algoritmos são também abordadas e definidas.
- O capítulo 5 apresenta questões da metodologia de funcionamento, desenvolvimento do simulador e a aplicação do método.
- O capítulo 6 apresenta os experimentos realizados e os resultados obtidos.
- O capítulo 7 apresenta a conclusão do trabalho e propostas para trabalhos e estudos futuros.

# Capítulo 2

## FUNDAMENTAÇÃO TEÓRICA

---

### 2.1 Filtro de Kalman

Filtro de Kalman é um algoritmo que utiliza uma série de medições observadas ao longo do tempo, contendo o ruído (variações geradas aleatoriamente pelo sistema), outras imprecisões e falta de dados, produzindo estimativas de variáveis desconhecidas que tendem a gerar informações mais precisas do que aquelas com baseados em uma medição simples.

O filtro foi desenvolvido inicialmente tentando resolver o problema do Filtro de Wiener (WIENER, 1949), formalmente o filtro de Kalman opera recursivamente em fluxos de dados de entrada ruidosos para produzir uma estimativa estatisticamente ótima do próximo estado do sistema (KALMAN, 1960) (GREWAL e ANDREWS, 1993).

Segundo Bishop (2006), as equações específicas para as atualizações das medidas e do tempo estão representadas nas Tabelas 2-1 e 2-2.

Pode-se notar como as equações de atualização de tempo na Equação 2.1 projeta as estimativas de estados atuais e covariância para a frente a partir do passo  $k-1$  até o passo  $k$ . As condições iniciais para o filtro serão discutidas nas referências anteriores.

**Tabela 2-1** - Filtro de Kalman Discreto atualização temporal.

$$\hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} \quad (2.1)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^t + \mathbf{Q} \quad (2.2)$$

**Tabela 2-2** - Filtro de Kalman Discreto atualização das medições.

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^t (\mathbf{H}\mathbf{P}_k^- \mathbf{H}^t + \mathbf{R})^{-1} \quad (2.3)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \quad (2.4)$$

$$\mathbf{P}_k = (\mathbf{1} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- \quad (2.5)$$

Sendo que:

$\hat{\mathbf{x}}$  = Estado anterior.

$\mathbf{P}_k^-$  = Covariância do erro prévio.

$\mathbf{K}_k$  = Ganho do filtro.

$\hat{\mathbf{x}}_k$  = Estado estimado.

$\mathbf{P}_k$  = Covariância do erro estimado.

$\mathbf{A}$  = Matriz de Transição de Estados.

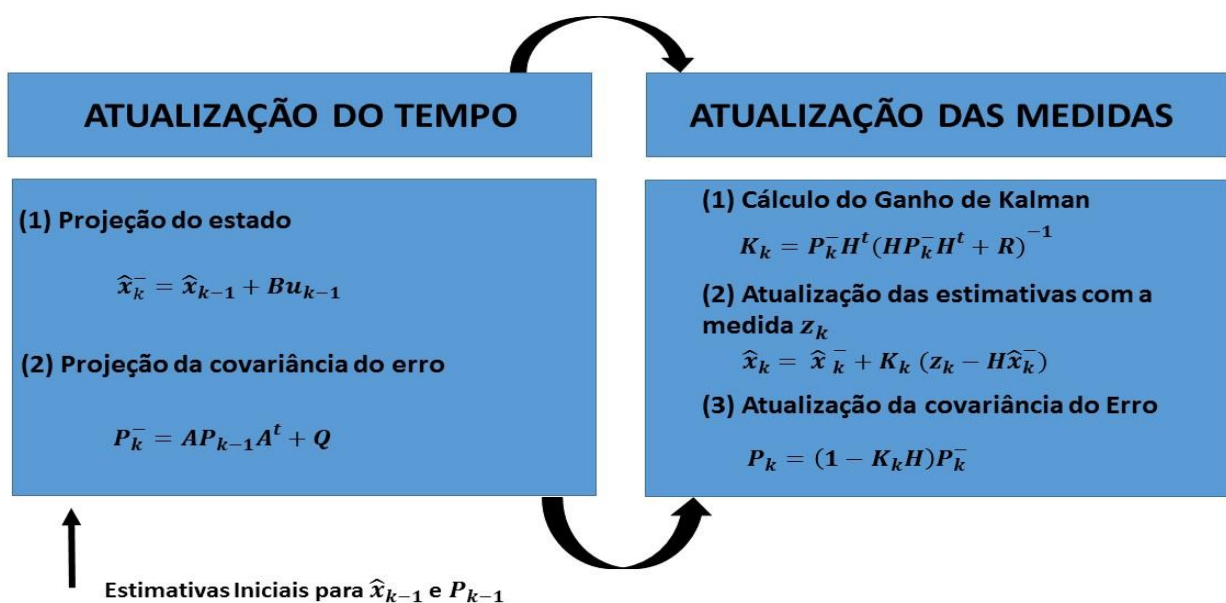
$\mathbf{B}$  = Matriz do Controle das Entradas.

$\mathbf{H}$  = Matriz de Dados das Medidas.

$\mathbf{Q}$  = Matriz de Covariância do Ruído.

$\mathbf{R}$  = Matriz de Covariância do Ruído das Medições.

A primeira tarefa durante a atualização da medição é calcular o ganho de Kalman com a equação dada em Equação 2.3. O próximo passo é realmente medir o processo e em seguida gerar uma estimativa de um estado posterior, incorporando a medida como em Equação 2.4. O passo final é a obtenção da covariância do erro para a posterior estimativa de covariância via Equação 2.5, como pode ser visualizado na Figura 2-1.



**Figura 2-1 - Operação do Filtro de Kalman**  
Adaptado de Bishop (2006).

Após cada período de atualizações e medições, o processo é repetido com o dado anterior, as estimativas posteriores são usadas para projetar ou prever as novas estimativas. Esta natureza recursiva é uma das características mais atraentes do Filtro de Kalman, por fazer implementações práticas muito mais simples que outros filtros que tendem utilizar todos os dados coletados para estimativas posteriores, com isso tendo grande custo computacional para cada estimativa.

## 2.2 Filtro de Kalman Estendido (EKF).

Como descrito na seção acima, o Filtro de Kalman aborda o problema geral de tentar estimar o estado de um processo controlado em tempo discreto, que é regulada por equações lineares de diferença estocástica, mas o que pode ser feito se o processo a ser controlado tem características não lineares, tempos atrás após intensa pesquisa foi construída umas das mais bem-sucedidas alterações do filtro de Kalman.

O filtro de Kalman que lineariza sobre a média e covariância atualmente é conhecida como Filtro de Kalman Estendido ou EKF (HAYKIN, 2001) (GREWAL e ANDREWS, 1993).

Conforme Bishop (2006), descreve em seu trabalho um processo a ser estimado com funções de diferença não lineares, podendo ser escrita com novas equações que podem linearizar estas funções, sendo estas descritas em nas Equações 2.6 e 2.7.

$$x_k = \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + W w_{k-1}, \quad (2.6)$$

$$z_k = \tilde{z}_k + H(x_k - \tilde{x}_k) + V v_k. \quad (2.7)$$

Sendo que:

$x_k$  and  $z_k$  são o estado atual e os vetores da medição.

$\tilde{x}_k$  and  $\tilde{z}_k$  são o estado aproximado e os vetores das equações.

$\tilde{x}_k = f(\hat{x}_{k-1}, u_{k-1}, 0)$  e  $\tilde{z}_k = h(\tilde{x}_k, 0)$ .

$\hat{x}_k$  é a próxima estimação de estado a cada passo  $k$ .

$w_k$  and  $v_k$  são variáveis randômicas que representam os ruídos do processo e das medições.

A é a Matriz Jacobiana das derivadas parciais de  $f$  com domínio em  $x$  sendo descrita pela Equação 2.8.

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}), \quad (2.8)$$

W é a Matriz Jacobiana das derivadas parciais de  $f$  com domínio em  $w$  sendo descrita pela Equação 2.9.

$$W_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}), \quad (2.9)$$

H é a Matriz Jacobiana das derivadas parciais de  $h$  com domínio em  $x$  sendo descrita pela Equação 2.10.

$$H_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\tilde{x}_k, \mathbf{0}), \quad (2.10)$$

V é a Matriz Jacobiana das derivadas parciais de  $h$  com domínio em  $v$  sendo descrita pela Equação 2.11.

$$V_{[i,j]} = \frac{\partial f_{[i]}}{\partial v_{[j]}}(\tilde{x}_k, \mathbf{0}), \quad (2.11)$$

Pode-se notar que para simplificação da notação não foi utilizado a variável temporal para cada etapa  $k$  com as matrizes jacobianas A, W, H e V sendo estas diferentes a cada etapa da interação temporal. Conforme demonstrado na Figura 2-2.

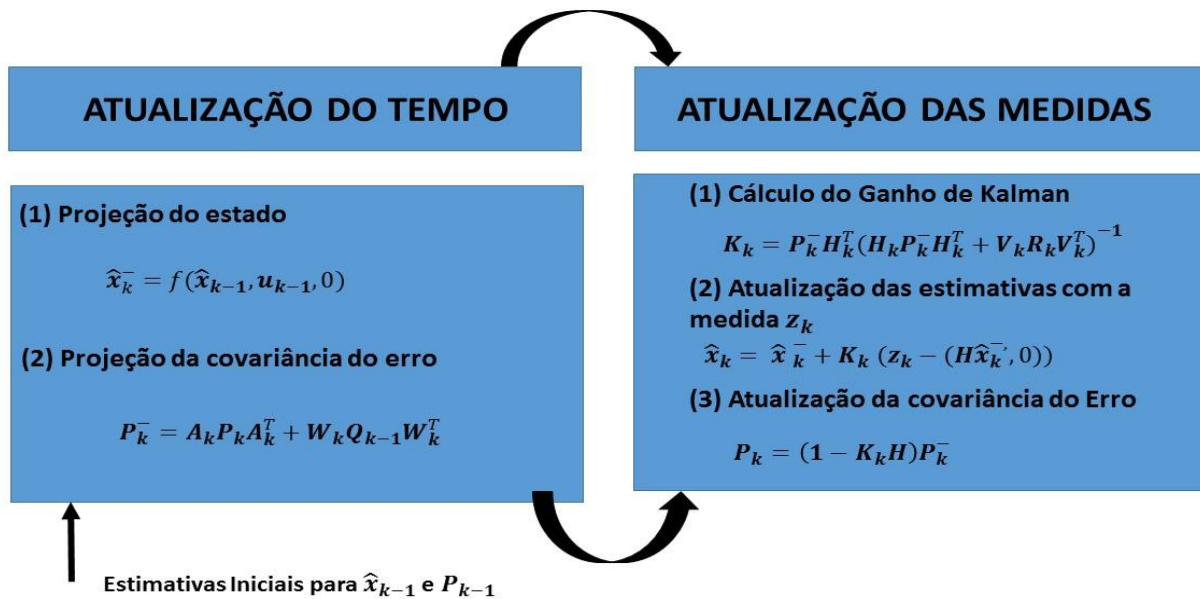


Figura 2-2 - Operação do Filtro Estendido  
 Adaptado de Bishop (2006).

### 2.3 Filtro de Kalman Unscented (UKF)

Os autores Julier e Uhlmann (1997), demonstraram os ganhos de desempenho substanciais do UKF no contexto da estimação de estados para o controle não-linear. Foi estendido o uso do UKF, a uma classe mais ampla de problemas de estimação não lineares, incluindo a identificação de sistema não lineares e no Treinamento de Redes Neurais.

A distribuição dos estados é representada por uma GRV (*Gaussian Random Variable*), mas agora é especificada usando um conjunto mínimo de pontos de amostragem cuidadosamente escolhidos. Estes pontos de amostra têm como objetivo a captura da média real e covariância do GRV (WAN e VAN DER MERWE, 2000).

As amostras não são sorteadas aleatoriamente, mas sim de acordo com um algoritmo determinístico. De modo que os problemas de convergência estatística não

tendem a diminuir a informação de alta ordem sobre a distribuição por ser capturada utilizando apenas um muito pequeno número de pontos.

As variáveis aleatórias  $X$  de tamanho  $n$  com médias  $\bar{x}$  e covariância  $P_{xx}$  são aproximadas por  $2n + 1$  pontos ponderados dadas por:

$$\begin{aligned} X_0 &= \bar{x} & W_0 &= k/(n + k) \\ X_i &= \bar{x} + (\sqrt{p(n + k)P_{xx}})_i & W_i &= 1/2(n + k) \\ X_{i+n} &= \bar{x} - (\sqrt{p(n + k)P_{xx}})_i & W_{i+n} &= 1/2(n + k) \end{aligned} \quad (2.12)$$

Quando  $k \in \mathfrak{R}$ ,  $(\sqrt{p(n + k)P_{xx}})_i$  é  $i$ -ésima linha ou coluna da matriz da raiz quadrada de  $(n + k)P_{xx}$  e  $W_i$  é o peso que esta associado com o  $i$ -ésimo ponto. (JULIER e UHLMANN, 1997)

O procedimento de transformação é o seguinte:

1. Instanciar cada ponto por meio da função para se obter o conjunto de pontos sigma transformados;

$$Y_i = f[X_i]. \quad (2.13)$$

2. A média é dada pela média ponderada dos pontos transformados;

$$\bar{y} = \sum_{i=0}^{2n} W_i Y_i \quad (2.14)$$

3. A covariância é outro produto ponderado dos pontos transformados;



$$\mathbf{P}_{yy} = \sum_{i=0}^{2n} W_i \{\mathbf{Y}_i - \bar{\mathbf{y}}\} \{\mathbf{Y}_i - \bar{\mathbf{y}}\}^T \quad (2.15)$$

Equações do UKF: (WAN e VAN DER MERWE, 2000)

- Inicializar com:

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0] \quad (2.16)$$

$$\mathbf{P}_0 = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] \quad (2.17)$$

$$\hat{\mathbf{x}}_0^a = E[\mathbf{x}^a] = [\hat{\mathbf{x}}_0^T \ 0 \ 0]^T \quad (2.18)$$

$$\mathbf{P}_0^a = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] = \begin{pmatrix} P_0 & 0 & 0 \\ 0 & P_v & 0 \\ 0 & 0 & P_n \end{pmatrix} \quad (2.19)$$

Para  $k \in \{1, \dots, \infty\}$ ,

- Calcule os Pontos Sigma:

$$\mathbf{X}_{k-1}^a = [\hat{\mathbf{x}}_0^a \ \hat{\mathbf{x}}_0^a \pm \sqrt{(L + \lambda)\mathbf{P}_{k-1}^a}] \quad (2.20)$$

- Atualização Temporal:

$$\mathbf{X}_{k|k-1}^x = \mathbf{F}[\mathbf{X}_{k-1}^x, \mathbf{X}_{k-1}^v] \quad (2.21)$$

$$\hat{\mathbf{x}}_k^- = \sum_{i=0}^{2L} W_i^{(m)} X_{i,k|k-1}^x \quad (2.22)$$

$$\mathbf{P}_k^- = \sum_{i=0}^{2L} W_i^{(c)} [X_{i,k|k-1}^x - \hat{\mathbf{x}}_k^-][X_{i,k|k-1}^x - \hat{\mathbf{x}}_k^-]^T \quad (2.23)$$

$$\mathbf{Y}_{k|k-1} = \mathbf{H}[\mathbf{X}_{k|k-1}^x, \mathbf{X}_{k-1}^n] \quad (2.24)$$

$$\mathbf{y}_k^- = \sum_{i=0}^{2L} W_i^{(m)} \mathbf{Y}_{i,k|k-1} \quad (2.25)$$

- Atualização das Medidas:

$$\mathbf{P}_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} = \sum_{i=0}^{2L} W_i^{(c)} [\mathbf{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-] [\mathbf{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \quad (2.26)$$

$$\mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} = \sum_{i=0}^{2L} W_i^{(c)} [\mathbf{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-] [\mathbf{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-]^T \quad (2.27)$$

$$\mathbf{K} = \mathbf{P}_{\mathbf{x}_k \mathbf{y}_k} \mathbf{P}_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k}^{-1} \quad (2.28)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}(\mathbf{x}_k \hat{\mathbf{y}}_k^-) \quad (2.29)$$

$$\mathbf{P}_k = \mathbf{P}_k^- - \mathbf{K} \mathbf{P}_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} \mathbf{K}^T \quad (2.30)$$

Quando  $\mathbf{x}^a = [\mathbf{x}^T \mathbf{v}^T \mathbf{n}^T]^T$ ,  $\mathbf{X}^a = [(\mathbf{X}^x)^T (\mathbf{X}^v)^T (\mathbf{X}^n)^T]^T$ .

Sendo:

$\lambda$  = Parâmetro de Escala Composta.

$L$  = Dimensão do Estado Expandido.

$\mathbf{P}_v$  = Covariância do Ruído de Processo.

$\mathbf{P}_n$  = Covariância do Ruído das Medidas.

$W_i$  = Distâncias calculada pela Equações 2.12.

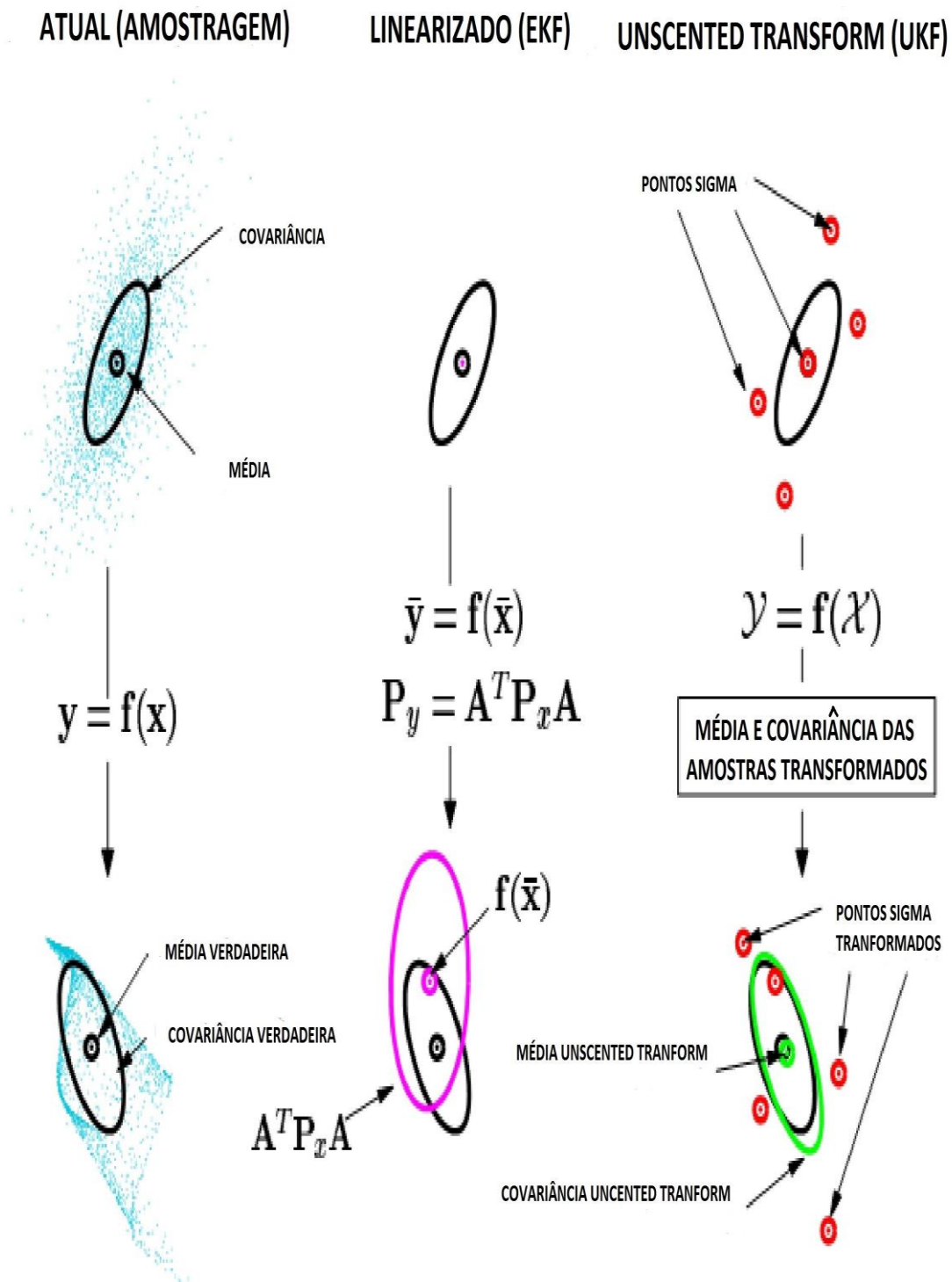


Figura 2-3 Representação dos Filtros EKF e UKF.

Adaptado de (WAN e VAN DER MERWE, 2000)

## 2.4 Suavizador Rauch Tung Striebel

O procedimento é baseado principalmente em um algoritmo de duas passagens chamado Rauch - Tung- Striebel (RTS), que tem por base de um passo para frente utilizando uma das variantes do Filtro de Kalman e uma recursão, suavizando os dados. Podendo evitar o ruído no processo de estimação da matriz covariância (H. E. RAUCH, STRIEBEL e TUNG, 1965) (HSIAO, 2011) (HAYKIN, 2001) (BAR-SHALOM, LI e KIRUBARAJAN, 2001).

Podendo ser utilizado para calcular a solução do suavizador para o modelo dado como uma distribuição, sendo descritas em Hartikainen e Särkkä (2008):

$$p(\mathbf{x}_k | \mathbf{y}_{1:T}) = N(\mathbf{x}_k | \mathbf{m}_k^s, \mathbf{P}_k^s) \quad (2.31)$$

A média e a covariância  $\mathbf{m}_k^s$  e  $\mathbf{P}_k^s$  são calculadas com as seguintes equações:

$$\mathbf{m}_{k+1}^- = \mathbf{A}_k \mathbf{m}_k \quad (2.32)$$

$$\mathbf{P}_{k+1}^- = \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_k \quad (2.33)$$

$$\mathbf{C}_k = \mathbf{P}_k \mathbf{A}_k^T [\mathbf{P}_{k+1}^-]^{-1} \quad (2.34)$$

$$\mathbf{m}_k^s = \mathbf{m}_k + \mathbf{C}_k [\mathbf{m}_{k+1}^s - \mathbf{m}_{k+1}^-] \quad (2.35)$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{C}_k [\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-] \mathbf{C}_k^T, \quad (2.36)$$

Onde:

$\mathbf{m}_k^s$  e  $\mathbf{P}_k^s$  são as estimativas suavizadas para a média e covariância na etapa de tempo  $k$ ;

$\mathbf{m}_k$  e  $\mathbf{P}_k$  são as estimativas do filtro para o estado média e covariância estado na etapa de tempo  $k$ ;

$\mathbf{m}_{k+1}^-$  e  $\mathbf{P}_{k+1}^-$  são o estado covariância significativo e estado previsto no passo de tempo  $k + 1$ , as mesmas executadas no Filtro de Kalman;

$\mathbf{C}_k$  é o ganho suavizado em tempo no passo  $k$ , que informa o quanto as estimativas suavizadas devem ser corrigidas nesse intervalo de tempo.

A diferença entre o Filtro de Kalman e sua versão suavizada, na versão suavizada ocorre recursividade para a frente e para trás de forma suave, como pode ser visto a partir das equações acima descritas.

A recursão suavizada tem início a partir do último passo de tempo  $T$  com  $\mathbf{m}_T^s = \mathbf{m}_T$  e  $\mathbf{P}_T^s = \mathbf{P}_T$ .

## 2.5 Redes Neurais

O estudo de Redes Neurais tem sido motivado pelas diferenças entre o funcionamento do cérebro humano e o de um computador digital tradicional. O cérebro humano é um sistema de processamento altamente complexo, não-linear e paralelo por natureza. A organização do cérebro permite que sejam realizadas certas computações (como o reconhecimento de padrões, o controle sensório-motor e a percepção) de maneira mais rápida e precisa que os mais poderosos computadores.

Por exemplo, uma pessoa consegue reconhecer um rosto familiar em uma cena não familiar demorando apenas uma fração de segundo, enquanto um computador convencional levaria horas ou dias para resolver uma versão simplificada desse problema.

A razão dessa grande capacidade do cérebro é a habilidade de desenvolver suas próprias regras, moldando os neurônios e criando o que é chamado de "experiência". Os neurônios possuem grande plasticidade, o que permite que o cérebro em desenvolvimento se adapte ao ambiente.

As redes neurais utilizam características de adaptação do cérebro humano e podem ser consideradas máquinas para modelar a maneira como o cérebro aprende uma tarefa ou função.

As RN são constituídas de neurônios artificiais e podem ser construídas com componentes eletrônicos, ou simuladas com computadores digitais. As RN alcançam bom desempenho através da interligação maciça de neurônios artificiais, utilizando um processo de aprendizagem sobre eles.

Uma RN pode ser definida como um processador maciçamente distribuído e paralelo, constituído de unidades simples de processamento, com a propensão natural de transformar conhecimento experimental em conhecimento pronto para uso (HAYKIN, 2001a).

As RN assemelham-se ao cérebro no sentido em que o conhecimento é adquirido do ambiente pela rede a partir do seu processo de aprendizagem e pela existência das forças de conexão entre os neurônios (pesos sinápticos), utilizadas para armazenar o conhecimento adquirido.

O procedimento pelo qual as RN aprendem é chamado de algoritmo de aprendizagem e serve para modificar os pesos sinápticos da rede, treinando a rede para que a mesma trabalhe (reconhecendo os padrões) da forma para a qual foi projetada. O aprendizado por modificação dos pesos é a forma tradicional pela qual as RN são projetadas.

As RN são comumente classificadas de várias formas: como subespecialidade da inteligência artificial; como uma classe de modelos matemáticos para classificação e reconhecimento de padrões; como parte da teoria conexionista de estados mentais ou como categoria de modelos em ciência da cognição (KOVÁCS, 2002).

Embora as RN sejam relacionadas com todas essas categorias, seria muito limitante classificá-las em apenas um desses setores. As RN formam hoje uma teoria genuína para o estudo de fenômenos complexos.

No que se refere à estrutura, as redes neurais possuem várias classificações:

- Relacionado ao número de camadas, as redes podem ser Redes de Camada Única ou Redes de Múltiplas Camadas (com a existência de uma ou mais camadas ocultas, ou intermediárias);

- Relacionado à conectividade, as redes podem ser totalmente conectadas ou parcialmente conectadas. Em uma rede totalmente conectada, cada neurônio possuirá ligações com todos os neurônios da camada seguinte. Quando a rede for parcialmente conectada, algumas dessas conexões não existirão. Em grande parte das aplicações as redes apresentam conectividade alta (totalmente ou quase totalmente conectadas);

- Relacionado à maneira como os sinais se propagam dentro da rede (se a rede possui retroalimentação ou não) as RN podem ser classificadas em alimentadas adiante ou recorrentes. Nas redes alimentadas adiante, o fluxo de sinal é apenas em um sentido, porém nas redes recorrentes, existe pelo menos um ciclo de retroalimentação, em que o sinal retorna para uma camada anterior. As redes recorrentes são muito utilizadas em processamento temporal, em que a retroalimentação serve para armazenar entradas de tempos anteriores e colocá-las novamente na entrada nos instantes seguintes.

### **2.5.1 Conceitos**

Uma RN pode ser caracterizada por três aspectos principais:

- O padrão de conexões entre as unidades (arquitetura);
- O método de determinação dos pesos das conexões (algoritmo de treinamento ou aprendizado);
- Função de ativação.

### **2.5.2 Importância das Redes Neurais**

As RN possuem a capacidade de generalizar informações, calculando saídas adequadas para entradas que não estavam presentes no arquivo de treinamento, ainda estão distantes de simularem um cérebro humano inteiro e trabalham apenas com subconjuntos de tarefas.

Mesmo assim, atualmente as redes neurais já se apresentam com grande destaque nas atividades que eram há pouco tempo essencialmente do cérebro e geram expectativa de grandes avanços nas próximas décadas. As principais vantagens que o uso de RN possibilita são descritos em Haykin (2001):

- **Não-linearidade**

As RN podem ter neurônios lineares ou não-lineares, a rede que possui ao menos um neurônio não-linear é considerada não-linear. A não-linearidade das RN é de um tipo especial, distribuída por toda a rede.

- **Mapeamento de Entrada-Saída**

As RN podem aprender através de exemplos, a partir de amostras rotuladas utilizadas no treinamento da rede. Assim a rede aprende com os exemplos a construir um mapeamento de entrada-saída para o problema considerado.

- **Adaptabilidade**

As redes possuem uma capacidade natural de adaptação dos pesos de seus neurônios de acordo com modificações no ambiente, podendo ser facilmente retreinadas.

Também existem projetos de redes que conseguem adaptar os seus pesos em tempo real, para trabalharem em ambientes não-estruturados.



- **Informação Contextual**

A informação contextual é tratada naturalmente por uma RN pois o conhecimento é representado pela sua própria estrutura. Cada neurônio é influenciado pela atividade dos outros, formando automaticamente a noção de contexto.

- **Tolerância a Falhas**

Se um neurônio ou suas conexões falharem (em implementações de redes físicas) a rede apresenta apenas uma degradação suave, devido à natureza distribuída da informação na rede.

- **Uniformidade de Análise e Projeto**

Os neurônios são os processadores universais de informação nas RN. Com isso é possível o compartilhamento de algoritmos de aprendizagem em diferentes aplicações de RN. Também podem ser construídas redes a partir de vários módulos.

- **Analogia Neurobiológica**

O estudo em RN é motivado pela analogia com o cérebro humano. O cérebro é uma grande prova de que o processamento paralelo, tolerante a falhas e adaptativo é, além de possível, muito rápido e poderoso.

As pesquisas visam desde auxiliar as ciências humanas e da saúde no entendimento dos fenômenos cerebrais, até desenvolver idéias para resolver

problemas mais complexos que os resolvidos por técnicas tradicionais, auxiliando as ciências exatas e da tecnologia.

### 2.5.3 Funções de Ativação

Abaixo são listadas algumas funções de ativação e/ou propagação que podem ser utilizadas na construção de Redes Neurais descritos em Zuben (2001). Estas funções podem ser vistas na Figura 2.5 e possuem as seguintes definições:

- Degrau (A):

$$f(x) = (1 \text{ se } x \geq 0 \quad 0 \text{ se } x < 0);$$

- Tangente hiperbólica logística (B):

$$f(x) = \frac{2}{1 + e^{-2x}} - 1;$$

- Linear logística (C):

$$f(x) = \begin{cases} -\log(1 - x) & \text{se } x \leq 0 \\ x & \text{se } 0 < x \leq 1 \\ 1 + \log & \text{se } x > 1; \end{cases}$$

- Sigmoide logística (D):

$$f(x) = \frac{1}{1 + e^{-x}} ;$$

- Linear pura (E):

$$f(x) = x;$$

- Base radial (F):

$$f(x) = e^{-x^2} ;$$

- Linear saturada (G):

$$f(x) = \begin{cases} 0 & \text{se } x \leq 0 \\ x & \text{se } 0 < x \leq 1 \\ 1 & \text{se } x > 1; \end{cases}$$

- Base triangular (H):

$$f(x) = \begin{cases} 1 - |x| & \text{se } -1 \leq x \leq 1 \\ 0 & \text{caso contrário.} \end{cases}$$

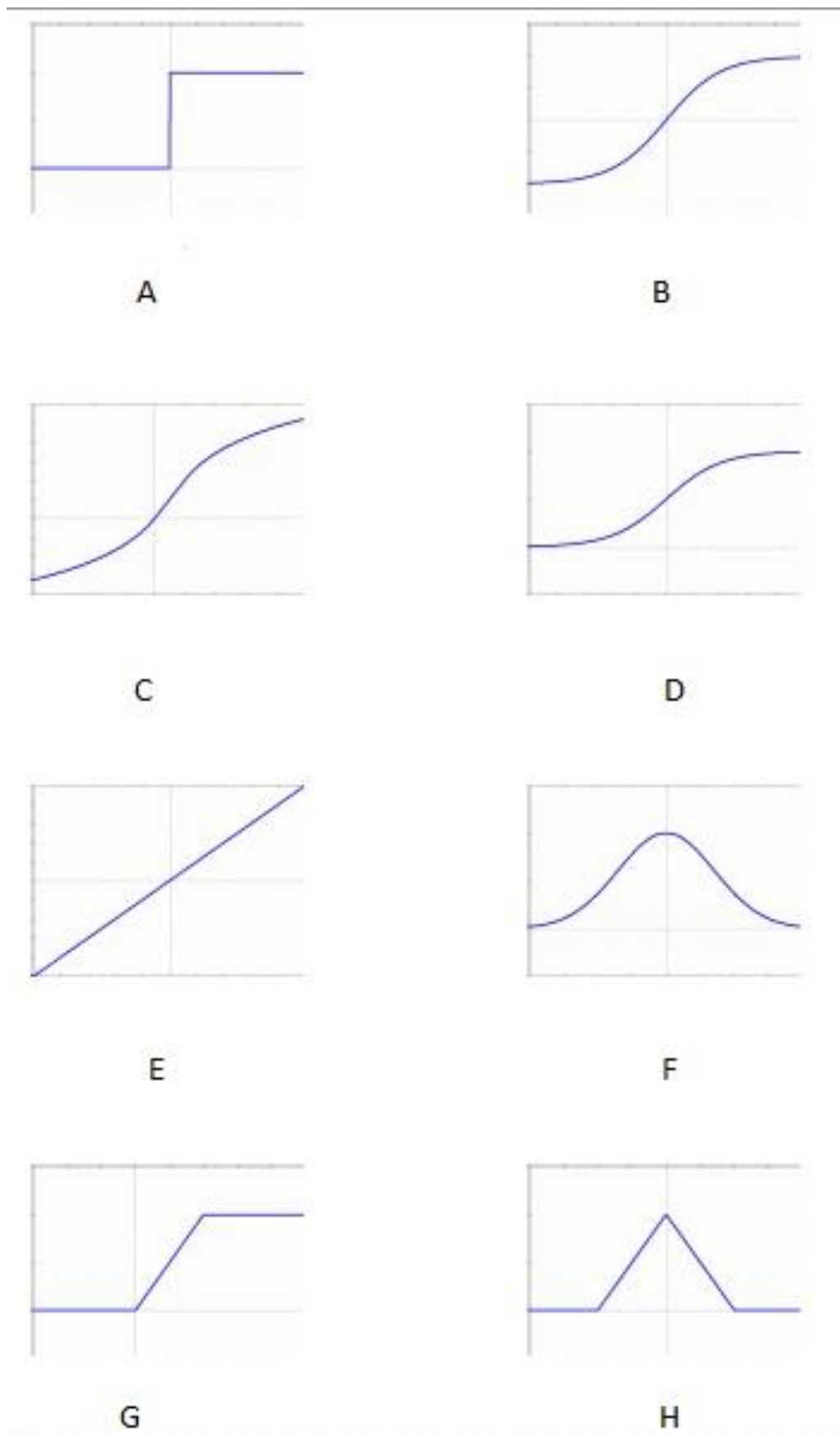


Figura 2-4 Funções de Transferência (ZUBEN, 2001).

### 2.5.4 Treinamento e Arquiteturas de RN

Os modelos neurais artificiais oferecem um paradigma atrativo, pois “aprendem” a resolver problemas através de exemplos, descritos por Mitchel (1997) e Zuben (2001), o treinamento de RN's pode ser dividido em:

- **Supervisionado:** Em um aprendizado supervisionado, a Rede Neural é treinada com auxílio de um supervisor. Para tanto, a rede deverá possuir pares de entrada e saída, ou seja, um conjunto de entradas e um conjunto com as saídas desejadas para cada entrada. Toda vez que for apresentada à rede uma entrada, deverá ser verificado se a saída obtida (gerada a partir dos cálculos efetuados a partir dos pesos que a rede possui) confere com a saída desejada para àquela entrada. Sendo diferente, a rede deverá ajustar os pesos de forma que armazene o conhecimento desejado. Esta interatividade do treino deverá ser repetida com todo conjunto de treinamento (entradas e saídas), até que a taxa de acerto esteja dentro de uma faixa considerada satisfatória;

- **Não-supervisionado:** quando não existe um agente externo indicando a resposta desejada para os padrões de entrada, este tipo de aprendizado também é conhecido como aprendizado auto supervisionado ou de auto-organização por que não requer saída desejada e/ou não precisa usar supervisores para seu treinamento. A seguir apresentamos uma lista com algumas das arquiteturas mais conhecidas até hoje. A distinção principal entre os modelos citados refere-se ao tipo de treinamento.

#### **Treinamento não-supervisionado:**

##### **Redes recorrentes:**

- Grossberg Aditivo (AG)
- Adaptive Resonance Theory (ART1)
- Hopfield Simétrico e Assimétrico (DH/CH)
- Memória Associativa Bidirecional (BAM)

- Memória Associativa Temporal (TAM)
- Mapa Auto-organizável de Kohonen (SOM)
- Aprendizado Competitivo

#### **Redes somente com propagação positiva (*feedforward*):**

- Learning Matrix (LM)
- Driver-Reinforcement Learning (DR)
- Memória Associativa Linear (LAM)
- Counterpropagation (CPN)

#### **Treinamento Supervisionado:**

##### **Redes Recorrentes:**

- Máquina de Boltzmann (BM)
- Mean Field Annealing (MFA)
- Cascade Correlation Recorrente (RCC)
- Aprendizado Recorrente em Tempo Real (RTRL)
- Filtro de Kalman Recorrente (EKF)

##### **Redes somente com propagação positiva (*feedforward*):**

- Perceptron • Adaline, Madaline
- Retro-propagação – Backpropagation (BP)
- Máquina de Cauchy (CM) • Artmap
- Rede Lógica Adaptativa (ALN)
- Cascade Correlation (CasCor)
- Filtro de Kalman (EKF)
- Filtro de Kalman (UKF)
- Learning Vector Quantization (LVQ)
- Rede Neural Probabilística (PNN)

## 2.6 Bearing Only

O problema básico na metodologia *Bearing Only Tracking* é tentar estimar a trajetória de um alvo com dados corrompidos por ruídos, sendo utilizado em problemas como:

Rastreamento Submarino (Sonar Passivo);

Vigilância Aeronáutica (Radar Passivo);

Rastreamento de Objetos (Ultrassom, Doppler, Laser).

Em um problema do sensor um único sensor, os dados (ângulos) são obtidos a partir de um objeto em movimento (BAR-SHALOM, LI e KIRUBARAJAN, 2001).

O alvo ou objeto está localizado nas coordenadas  $(x^t, y^t)$ , se movendo com uma velocidade constante no vetor  $(\dot{x}^t, \dot{y}^t)$ .

Conforme descrito em K.Radhakrishnan, Unnikrishnan e Balakrishnan (2010) o vetor de estado do alvo pode ser definido como:

$$X^t = [x^t, \dot{x}^t, y^t, \dot{y}^t]^T \quad (2.37)$$

Em que  $(x, y)$ , e  $(\dot{x}, \dot{y})$  são componentes de posição e de velocidade, o estado do observador pode ser definido da mesma forma como:

$$X^0 = [x^0, \dot{x}^0, y^0, \dot{y}^0]^T \quad (2.38)$$

O vetor do estado relativo é definido como sendo:

$$X_1 = X^T - X^0 = [x, \dot{y}, x, \dot{y}]^T \quad (2.39)$$

Na prática, as formas mais comuns de rastreamento ou localização de objetos no Plano Cartesiano denotado como (1) na Equação 2.39, é utilizar modelos:

Modelo de Velocidade Constante (*Constant Velocity Model*);

Modelo de Curva Coordenada (*Coordinated Turn Model*);

Modelo de Velocidade de Wiener (*Wiener Constant Velocity*).

Os modelos citados podem ser melhor estudados em Bar-Shalom, Li e Kirubarajan (2001), K.Radhakrishnan, Unnikrishnan e Balakrishnan (2010) e Simo Särkkä (2004).



# Capítulo 3

## REVISÃO BIBLIOGRÁFICA

---

### 3.1 Revisão Bibliográfica

Diversos estudos ao longo dos anos, utilizaram diferentes abordagens para o uso de Redes Neurais em adição aos Filtros de Kalman. Em seus trabalhos Stubberud e Owen (1998), Stubberud, Lobbia e Owen (1995), descrevem o funcionamento de um Filtro de Kalman Estendido Neural Online, enquanto o controle esta sendo aplicado, faz-se uso de uma Rede Neural (RN) para prever o erro associado ao EKF.

Nesses documentos é demonstrada a eficácia de um Neuro Observador melhorado, com a atualização de estados preditivos otimizadas do EKF por uma RN, podendo esta RN ser considerada como um aproximador de função que rege o modelo, sendo utilizado desta forma em diversos trabalhos (BALDI e HORNIK, 1995).

Nos sistemas que utilizam EKF mostra-se necessário, em primeiro plano um conhecimento avançado do modelo em estudo, computando assim simultaneamente a previsão estimada e sua matriz Jacobiana. Quando estes parâmetros são desconhecidos ou inseridos incorretamente, ocorrem grandes erros na previsão do EKF.

A RN tende a aprender o funcionamento dinâmico do sistema, tentando assim obter uma melhor função que descreva o sistema, evitando assim erros na modelagem.

Em Haykin (2001) é demonstrado o uso do EKF com uma Rede Neural Multicamada Recorrente para modelar uma série caótica ou que poderia ser potencialmente caótica. Sendo que estes processos são comandados por um conjunto de sistemas de equações diferenciais acopladas, sendo sensível às condições iniciais, uma perturbação inicial no princípio do deslocamento provoca um deslocamento nas previsões não sistemático e não linear cumulativo durante o processo de amostragem.

Para a avaliação do modelo proposto, o autor utiliza a equação Lorenz, por se tratar de sistema caótico complexo de difícil modelagem. Para avaliação do algoritmo, o modelo *Signal Error Ratio* (SER), expresso em decibéis, é definido por

$$SER = 10 \log 10 \frac{MSS}{MSE} \quad (3.1)$$

Onde MSS é o Erro Quadrado Médio dos dados de testes reais e MSE é o Erro Médio da previsão. Algoritmos com melhor desempenho, como o filtro de UKF podem substituir o Filtro de Kalman Estendido, pois o Filtro EKF tem duas desvantagens potenciais importantes.

Em primeiro lugar, as derivações das matrizes Jacobianas, com suas aproximações lineares para as funções não-lineares podem ser complexas, causando dificuldades de implementação.

Em segundo lugar, estas linearizações podem levar a instabilidade dos dados, se os intervalos na coleta de dados forem demasiadamente curtos, para resolver essas limitações, Julier e Uhlmann desenvolveram o UKF, este filtro tornou-se uma alternativa viável para EKF para prever novos estados não-lineares.

A principal justificativa é mais fácil utilizar uma distribuição Gaussiana para aproximação de uma função não-linear arbitrária, em comparação a linearização usando matrizes Jacobianas, o UKF utiliza-se de uma abordagem de amostragem

determinística para capturar a média de variância com um conjunto mínimo de pontos de amostra (LIU, SHUI e LI, 2011), (LAVIOLA, 2013).

O "Neural Network-UKF" é superior ao "Neural Network-EKF" em diversas comparações. Sendo discutido também em Zhan e Wan (2006), que tem participação de uma RN, obtendo maior capacidade de predição, comparado respectivamente com o EKF ou UKF, especialmente na presença de incertezas. Para comparação é usada a aproximação da Equação não-linear 3.2:

$$y = \frac{2}{1 + \exp\left(\frac{1}{1 + \exp(0.1 - 0.5x)} + \frac{1}{1 + \exp(0.5 + 0.4x)} - 1\right) + 0.5 \sin(0.5x) + 0.5 \frac{x}{1+x^2}} \quad (3.2)$$

A Tabela 3.1 demonstra a comparação entre os métodos, se utilizando de erro e covariância como parâmetros qualitativos.

**Tabela 3-1** - Comparação de diferentes modificações (ZHAN e WAN, 2006).

Algoritmos	Erro médio quadrático (MSE)	
	Média	Variância
EKF	0.3584	0,01295
UKF	0.2661	0,00925
NN-EKF	0.1380	0,00696
NN-UKF	0.0769	0,00176

Comparações sobre EKF e UKF algoritmos são discutidos por Kurt e Yavuz (2012), Mallick, Morelande e Mihaylova (2012) e D'alfonso Lucia, *et al.* (2011), mas ao longo do tempo vieram outras variantes do Filtro de Kalman tentando propor melhorias

em vários aspectos, tais como velocidade de resposta, precisão e custo computacional.

### **3.2 Dentre os algoritmos propostos, o Cubature Kalman Filter (CKF) utilizando a metodologia matemática conhecida como regra de Cubagem Esférica Radial.**

Basicamente derivando o terceiro grau da regra cubagem esférica radial, fornecendo assim um conjunto de pontos de cubagem de escala linear com a dimensão do estado vetor (HAYKIN, 2001).

O trabalho de Crouse (2013), traz grandes avanços estendendo o filtro CKF, a fim de comparar os Filtros UKF e CKF tentando determinar uma melhor forma de obter a melhor resposta do algoritmo CKF Estendido.

Tendo usado um modelo não-linear, um melhor desempenho do filtro CKF de quinta ordem ou maior foi obtido, demonstrando uma maior precisão e consistência na abordagem do modelo em ambientes com alto ruído angular (cenário difícil), em comparação com o UKF.

Traçando um paralelo com o trabalho descrito acima Jia, Xin e Cheng, (2012), traz para o seu trabalho uma nova proposta para o filtro de alta ordem CKF, obtendo excelentes resultados como demonstrado nas Figuras 3-1 e 3-2.

Os resultados representativos obtidos em Crouse (2013) e Jia, Xin e Cheng (2012), sendo motivação para a proposta do trabalho demonstrada posteriormente na proposta deste trabalho, a análise inicial de algoritmos EKF, UKF e CKF.

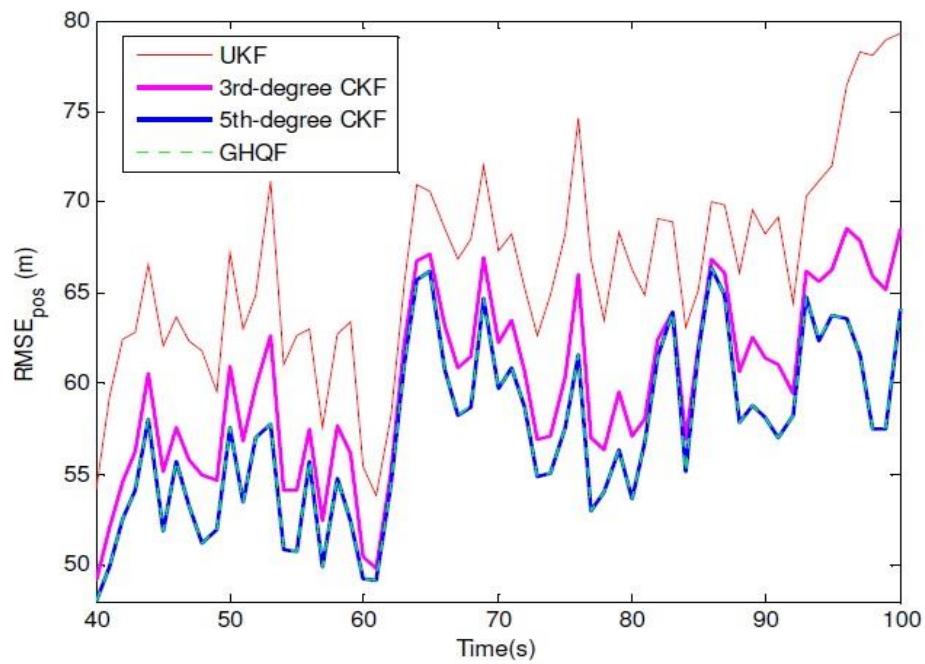


Figura 3-1 - Erro Médio Quadrático da Posição.

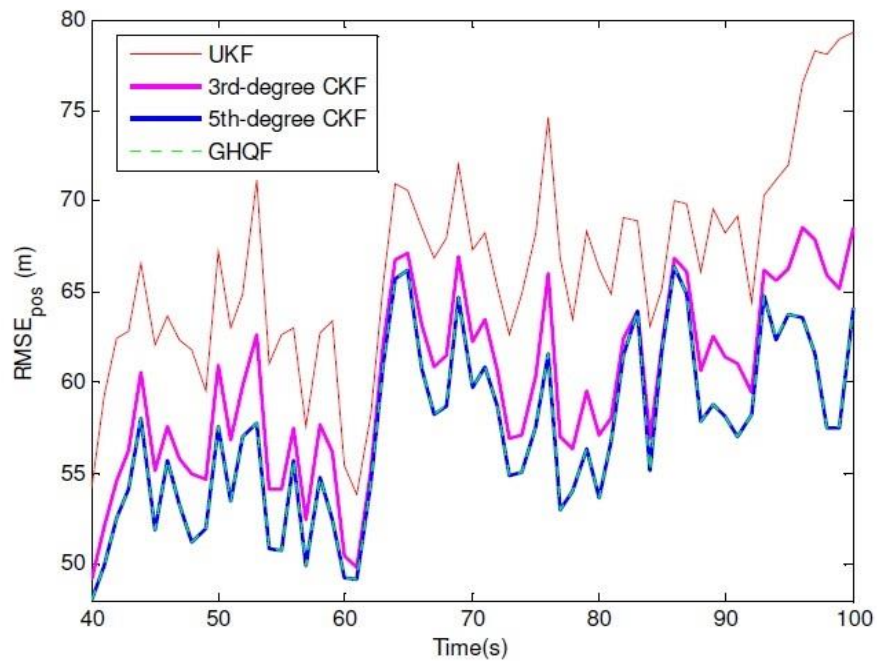


Figura 3-2 - Erro Médio Quadrático da Velocidade.

Faz-se necessário o uso do Filtro CKF no estudo preliminar deste trabalho, pois o mesmo obteve excelentes resultados no aumento de performance e

acuracidade, utilizando de Redes Neurais como assistente na determinação do modelo estudado, esta implementação pode ser melhor entendida em sua implementação em Arasaratnam e Haykin (2008).

Os trabalhos citados abaixo, demonstram a busca de diversos autores em métodos para estruturação de ambientes, sistemas de posicionamento robótico e mapeamento.

Esse capítulo expõe os conceitos básicos e abordagens necessárias para entendimento do presente trabalho, pautando-se em diversas pesquisas e exemplificando de maneira mais detalhada os pontos motivadores da hipótese levantada, os trabalhos iniciais demonstram a busca por metodologias para as tarefas propostas nesse trabalho, a dificuldade encontrada por esses autores culminou na determinação e especificação da proposta apresentada no capítulo 4.

### **3.3 Trabalhos Correlacionados**

No breve resumo apresentado percebe-se claramente que o Filtro de Kalman e suas possíveis modificações, é um tema atualmente abordado e que sofre constante alterações e novas propostas, para os mais diversos objetivos. Porém existe na literatura um problema frequentemente abordado, a velocidade de computação dessas modificações, precisão e acurácia para sistemas com características caóticas ou potencialmente caóticas. Vários autores procuram alternativas viáveis para solução desses problemas, sendo evidenciado neste trabalho, as soluções para servoposicionamento nos mais diversos ambientes, reais ou simulados.

### **3.4 A Survey of Object Recognition Methods for Automatic Asset Detection in High-Definition Video (WARSOP, 2010).**

Para início das apresentações sobre os trabalhos que abordam reconhecimento de padrões em ambientes não estruturados, com diferentes motivações e tratamento de problemas, considera-se importante o trabalho efetuado por Warsop (2010), sendo este contemporâneo, com aprofundado levantamento bibliográfico atualizado sobre reconhecimento de padrões citando suas diversas metodologias e aplicações.

Demonstra também a utilização de vários métodos para análise comparativa e qualitativa, trazendo também diversas características impeditivas no uso dos mesmos, como fatores de iluminação, posição e distância.

O autor também conclui que todos os diferentes métodos citados em seu trabalho podem ser utilizados nas mais diferentes tarefas de reconhecimento em diversas distâncias possíveis.

Os métodos de escala invariante são aplicáveis pelo seguinte pressuposto, que os objetos se deslocam ao longo e a frente da câmera, podendo estes alterar seus tamanhos, os métodos de distância invariante estão intrinsicamente ligados aos métodos de escala invariante, o domínio a serem utilizadas estas metodologias dependem diretamente do domínio e distância a serem utilizados.

Os sistemas de visão ativa podem utilizar-se de câmeras dotadas de zoom, fazendo que os objetos possam ser captados em seu tamanho original.

### **3.5 Road sides recognition in non-structured environments by vision (AUFRKRE, AUFRKRE, *et al.*, 2004).**

Segundo Aufrkre, *et al.* (2004), as pesquisas relacionadas ao desenvolvimento de sistemas autoguiados ou auxiliados por sistemas GPS em ambientes não estruturados têm se tornando fonte de inúmeros trabalhos importantes nesta área de desenvolvimento.

O autor se utiliza de três importantes dados iniciais na fase de aquisição, estes são as informações de posição advindas do satélite, as imagens do sistema CCD e as variáveis referentes a inercia do sistema nos três eixos relacionais, sendo estes providos por acelerômetro acoplado ao sistema, este dados são pré-processados e enviados para dois algoritmos específicos, tendo como resultado uma interface homem máquina, sendo citado pelo autor que posteriormente esta saída será um controle de mobilidade robótico.

No trabalho se torna evidente que estes novos algoritmos, atualizados recursivamente tendem a buscar a situação ótima a cada nova interação, os grandes benefícios deste processo são a alta acurácia do sistema, a rápida detecção global e baixa sensibilidade a oclusões.



### **3.6 Movement-Flow-Based Visual Servoing and Force Control Fusion for Manipulation Tasks in Unstructured Environments (TORRES, 2005).**

O artigo desenvolvido por Torres (2005), apresenta uma nova abordagem para a fusão, das informações advindas de sensores visuais e sensores de força, e sua aplicação para tarefas de manipulação em ambientes não estruturados. Utilizando um sistema de Servovisão clássico baseado em imagem, em uma trajetória tridimensional 3D, o posicionamento inicial e as configurações desejadas não podem ser especificados, especialmente em casos de grandes diferenças de rotação e na maioria das aplicações em que é geralmente utilizada a metodologia ponto a ponto.

Devido à natureza diferente da magnitude medida com os sistemas visuais e de força, um grande número de abordagens vem sendo utilizadas para a fusão destas informações, obtendo assim um sistema de controle híbrido.

Diferentes fases que compõem a estratégia de fusão são aplicadas para a tarefa e são demonstradas de forma simples e didática no trabalho. Sendo que em uma determinada tarefa, o robô deve manter o contato plano com uma força constante que esta sendo aplicada entre a garra e a superfície. A Servovisão é baseada no fluxo de movimento e utilizado para que a trajetória da imagem desejada, sendo que cada característica de deslocamento é observada pela câmera durante a trajetória do robô (mantendo o contato com a superfície).

No entanto, essa trajetória está sujeita a erros e, portanto, deve ser modificado de acordo com as informações obtidas a partir do sensor de força.

O método descrito pelo autor foi aplicado a diferentes tarefas de manipulação que exigiam grande precisão. Neste tipo de aplicação, não é necessário somente obter características iniciais obtidas pela câmera, mas um sistema de acompanhamento da trajetória determinada entre os pontos da tarefa, cumprindo as restrições espaciais desejadas com auxílio do sensor de força, obtendo dessa maneira o correto desenvolvimento da tarefa.

### **3.7 Simple, Robust Autonomous Grasping in Unstructured Environments (DOLLAR e HOWE, 2007).**

A incerteza na relação entre o objeto e a garra torna difícil o controle das forças de contato, a tentativa de estabelecer um contato ativo que muitas vezes ocorre em tarefas de agarramento ou posicionamento sobre objetos em ambientes não estruturados, podem resultar em grandes forças de contato, a menos que a garra seja dedicada para este fim.

O contato com força em demasia ocorre devido à incerteza no posicionamento em ambientes não estruturados, mas também pode acontecer em experiências de laboratório, particularmente na fase de depuração. Os pesquisadores são muitas vezes relutantes na depuração, pelo risco de acidentes com garras de robô com vários graus de liberdade, por isso implementações devem ser cuidadosamente validadas e o escopo experimental deve ser limitado.

Empiricamente o autor visa demonstrar que podem ser otimizadas as movimentações das articulações passivamente, sendo assim o acoplamento adaptativo da garra pode permitir que o sistema se adapte a grandes erros de posicionamento, que podem ocorrer nesse tipo de tarefa. Para isto, é utilizado em conjunto com um modelo aproximado do tamanho do objeto e o local de detecção, tornando a tarefa agarrar ainda mais robusta a variações na forma do objeto e posição.

O autor ainda cita que graus de orientação adicionais, auxiliam na melhoria do desempenho através de uma melhor posição relativa ao objeto.

### **3.8 Probabilistic Scan Matching for Motion Estimation in Unstructured environments (MONTESANO, MINGUEZ e MONTANO, 2005).**

O objetivo do autor utilizando técnicas de varredura probabilística é calcular o movimento relativo do sensor entre duas posições consecutivas, maximizando a sobreposição entre as medições do sensor obtidos em cada nova posição. Estas técnicas têm sido amplamente utilizadas no contexto de mapeamento, registro de dados em 3D, reconhecimento de objetos ou estruturação ambiental.

Este artigo apresenta um algoritmo de digitalização com varredura probabilística para estimar o deslocamento planar de um robô por escaneamento de intervalos bidimensionais. A contribuição é uma modelagem probabilística deste processo que leva em conta a incerteza da localização do sensor e os ruídos do processo de medição. Isto aumenta a robustez do método em cenários reais em comparação a métodos geométricos puros.

Além disso, este sistema captura a translação e rotação simultaneamente, tendo como consequência, que o método final proposto é capaz de lidar com grandes erros de posicionamento especialmente em relação a rotação, sendo esta a dificuldade da maioria das abordagens existentes.

Os resultados deste trabalho demonstram que os métodos amplamente difundidos, podem melhorados quanto na precisão, robustez e convergência, utilizando novos cálculos, metodologias ou técnicas adicionais.

### **3.9 Appearance-Based Visual Learning in a Neuro-Fuzzy Model for Fine-Positioning of Manipulators (ZHANG, SCHMIDT e KNOLL, 1999).**

Tarefas relativamente simples são efetuadas na indústria com manipuladores do tipo garra, esse sistema propõe a adição de sistema de coleta de imagens para posicionamento ou coleção de objetos de variadas geometrias e localização espacial.

Sobre sistema de coleta de imagens e posicionamento existem características intrínsecas do processo para geração de falhas como por exemplo calibração do sistema, luminosidade do ambiente, sistema de treinamento e aprendizado

Sendo proposto que o manipulador robótico detecte simetrias básicas, sem treinamento assistido, sem calibração da câmera e sem necessidades de algoritmos específicos para cada geometria a ser detectada.

São aplicadas neste trabalho técnicas de processamento e análise de imagens para determinação de posição e geometria, em contrapartida utilizando técnicas de aprendizado de máquina, os mesmos obtêm um sistema robusto a variantes externas com características otimizadas como ausência de calibração e mapeamento direto a um modelo livre de contexto.

### **3.10 Distributed Multiple View Fusion for Two-Arm Distance Estimation (SCHEERING, ZHANG e KNOLL, 1999).**

Com o objetivo de usar um controlador com aprendizado supervisionado, utilizando a relação entre quatro vistas arbitrárias de uma câmera não calibrada e a distância de dois braços robóticos.

Para se obter isso, foi construído um Controlador *Neuro Fuzzy B Spline* tendo como função base a produção de um arco de controle dos vértices por uma abordagem em Redes Neurais. Geralmente redes neurais ou sistemas Fuzzy tem um grande número de variáveis entrada, e estas sofrem do problema erros intrínsecos relativos ao processo, como por exemplo luminosidade, velocidade de captura, qualidade das lentes, tratamentos matemáticos entre outros.

Por isso é essencial para reduzir os erros no processo reduzir o índice de perda de informações. Explorando a múltipla redundância entre as exibições, o sistema proposto permite calcular as distâncias mesmo sem presença de todas as sobreposições parciais. O modelo *B-Spline* serve como um eficiente e não-linear interpolador para interpretar regras de controle. As vantagens neste conceito estão dispostas abaixo:

O treinamento pode ser programado que as imagens mais representativas sejam quantificadas automaticamente. Esta característica, juntamente com a processamento paralelo introduzida no processo de fusão das imagens, torna o sistema robusto em o mais rápido possível aprox. 10 Hz.

Os parâmetros internos e externos da câmera não precisam ser conhecidos para estimar a distância, sendo assim a calibração das câmeras é supérfluo. Em um trabalho futuro o autor testará o algoritmo com pequenas mudanças de posição na câmera e alteração da luminosidade.

### **3.11 Targeted On-line Modeling for an Extended Kalman Filter Using Artificial Neural Networks (STUBBERUD e OWEN, 1998).**

O autor demonstra técnicas de implementação de um filtro de Kalman Estendido (EKF) sendo auxiliado por uma Rede Neural Artificial, com treinamento on-

line. A finalidade da RN é obter modelo dinâmico do sistema, que são usados no processo do EKF.

Usando uma RN que descreva todo o modelo que tem como alvo a dinâmica de estados específicos do sistema. A ideia é mostrar que a segmentação estados específicos irão reduzir os cálculos, mantendo um elevado grau de eficácia.

Os resultados preliminares obtidos no trabalho citado demonstram que a segmentação das capacidades de modelação do RN no EKF tem um efeito benéfico na medida em que pode reduzir o custo computacional do algoritmo. Embora estes resultados não são totalmente conclusivos, eles indicam que mais pesquisas no direcionamento dos erros do modelo do sistema é uma proposta com grandes propensões ao sucesso.

### **3.12 Filtros de Kalman e suas aplicações.**

Os trabalhos acima citados descrevem a pesquisa realizada por diversos autores tentando definir uma metodologia que obtenha bons resultados nas tarefas de posicionamento e ambientação estrutural.

Chen (2012) descreve diversos métodos utilizando Filtros de Kalman para estruturação ambiental e posicionamento, algumas delas estão abaixo descritas na Tabela 3-2.

Informações sobre os autores podem ser consultados no Trabalho de Pesquisa de Chen (2012), outros autores relevantes com diferentes metodologias, estão abaixo descritos, sendo estes trabalhos utilizados como base em nossa pesquisa.

**Tabela 3-2 - Trabalhos relevantes realizados utilizando filtros de Kalman.  
Adaptado de Chen (2012)**

<b>PROPÓSITO/TAREFA</b>	<b>MÉTODO</b>
<b>LOCALIZAÇÃO DE ROBÔS MOVEIS</b>	EKF para localização de Robôs
<b>NAVEGAÇÃO BASEADA EM VISÃO COMPUTACIONAL</b>	EKF para estimação da localização e orientação
<b>DETECÇÃO DE OBJETOS</b>	FK para integração da informação e adaptação de parâmetros
<b>ESTABILIZAÇÃO DE IMAGEM</b>	EKF para a predição de ângulo em tempo real
<b>LOCALIZAÇÃO EM TEMPO REAL</b>	EKF para estimação e otimização de parâmetros
<b>SERVO CONTROLE VISUAL</b>	UKF para fusão de dados  EKF com rastreador Multi Objetivos
<b>ESTIMAÇÃO DE POSIÇÃO</b>	EKF para compensação dos erros
<b>FUSÃO DE DADOS</b>	EKF/UKF

Sendo eles Arasaratnam e Haykin (2008), Auger, Hilairet, *et al.*, (2013), Baldi e Hornik (1995), Bar-Shalom, Li E Kirubarajan (2001), Chaumette e Hutchinson (2006), Chaumette e Hutchinson (2007), Crouse (2013), D'alfonso, Lucia, *et al.*, (2011), Haykin (2001), Hartikainen e Särkkä (2008), Hsiao (2011), Janabi-Sharifi e Marey (2010), Zhong, Zhong e Peng (2013), Khan Salman, *et al.*, (2015), Sun, Ma e Li (2015) e Wang e Deng (2015).

### **3.13 Considerações Finais**

A citação de alguns artigos e a breve revisão sobre os temas Redes Neurais, Servovisão e Filtros de Kalman demonstra a relevância das pesquisas acerca destes temas, sendo que surgem novas modificações, propostas e algoritmos para a solução de problemas característicos de cada um dos temas abordados.

Os resultados obtidos por esses diversos autores motivam e dão foco a essa proposta, pois a união cooperativa entre os Filtros de Kalman e Redes Neurais voltada ao servoposicionamento, podem trazer resultados interessantes, problemas ou soluções ainda não tratadas ou abordadas.



# Capítulo 4

## PROPOSTA DO TRABALHO

---

Esse trabalho apresenta uma proposta de desenvolvimento e testes de metodologias para posicionamento em ambientes não estruturados, aplicado em operações de rastreamento de objetos. Os testes serão realizados em simuladores e validado com diversos tipos de movimentos, afim de validar o modelo.

Os estudos, análises e testes devem possibilitar a conclusão se o método proposto, pode ou não ser aplicado em uma situação de rastreamento, principalmente no rastreamento de objetos com características de movimentação em ambientes de difícil percepção espacial.

O método a ser utilizado no teste das Séries e na implementação da Rede Neural, neste trabalho é baseado no livro “*Kalman Filtering and Neural Networks*” (HAYKIN, 2001) e na pesquisa sobre rastreamento e navegação escrito por Bar-Shalom, Li e Kirubarajan (2001), estas referências e metodologias são utilizadas para a avaliação do desempenho do método proposto.

Os algoritmos serão implementados utilizando a ferramenta MATLAB desenvolvida pela MathWorks ©, com a adição do Toolbox REBEL (WAN e VAN DER MERWE, 2000), Dynaest (BAR-SHALOM, LI e KIRUBARAJAN, 2001) e EKF/UKF (HARTIKAINEN e SÄRKKÄ, 2008).

A verificação e a validação do sistema serão realizadas utilizando a ferramenta MATLAB 2015 Student Version, sendo esta baseada nos trabalhos de Aufkre, *et al.* (2004), Bar-Shalom, Li e Kirubarajan (2001), Corke (2011), Simo Särkkä (2004), Stubberud, Lobbia e Owen (1995), Hartikainen e Särkkä (2008), Zhan e Wan (2006) e Zhang, Schmidt e Knoll (1999).

## 4.1 Redes Neurais

As redes neurais executam as previsões de funções complexas dos sistemas estudados. As RN aprendem a partir de amostras dos próprios dados, fazendo ajustes de maneira gradual, tentando obter a função geradora que melhor corresponda ao sistema proposto.

Com isso, as RN não necessitam conhecer previamente o modelo estatístico gerador das séries. As RN apresentam uma não-linearidade de um tipo especial: presente em cada um de seus neurônios.

A combinação de não-linearidades de vários neurônios de camadas ocultas ou de sucessivas camadas torna as RN muito poderosas, proporcionando o tratamento de altos graus de não-linearidades nas séries (HAYKIN, 1999).

A dificuldade existente nas RN encontra-se no fato de que elas, como não modelam ruído possam confundir o sinal (função original do sistema) com o ruído.

A intenção deste trabalho é unir a capacidade de modelagem de ruído (presente no Filtro de Kalman), com adaptação a modelos desconhecidos e tratamento de não-linearidades (presentes nas RN).

Foi considerado neste trabalho dois tipos de arquitetura de rede, sendo estes melhor discutidos Haykin (1999) e Haykin (2001): As redes sem realimentação (*feedforward*) e sua extensão dinâmica, as redes com realimentação ou recorrentes (*recurrent*).

## 4.2 Análise de Performance do Sistema Neural e Posicionamento

Na proposta para avaliação das metodologias e desempenho do sistema, será utilizada a ferramenta estatística, Erro Quadrático Médio (MSE) que segundo Lehmann e Casella (1998) é um estimador que utiliza a média dos quadrados dos "erros", ou seja, a diferença entre o estimador e o que se estima. MSE é uma função de risco, o que corresponde ao valor esperado do erro quadrado perda ou perda quadrática. A diferença ocorre devido a aleatoriedade ou porque o estimador não leva em conta a informação que poderia produzir uma estimativa mais precisa.

O MSE incorpora tanto a variância do estimador e quanto sua polarização. Para um estimador imparcial, o MSE é a variância do estimador. Tal como a variância, MSE tem as mesmas unidades de medida, como o quadrado da quantidade a ser estimado.

Em uma analogia com desvio padrão, tendo a raiz quadrada do MSE produz o erro da raiz quadrada média ou desvio da raiz quadrada média (RMSE ou RMSD), que tem as mesmas unidades que a quantidade a ser estimado; para um estimador imparcial, o RMSE é a raiz quadrada da variância, conhecida como o desvio padrão.

## 4.3 Testes de Posicionamento em Simulador

Nas ferramentas REBEL (WAN e VAN DER MERWE, 2000), EKF/UKF Toolbox (HARTIKAINEN e SÄRKKÄ, 2008) e Dynaest (BAR-SHALOM, LI e KIRUBARAJAN, 2001), existe uma vasta documentação sobre sistemas de filtragem e rastreamento de objetos com possibilidade de alteração nos códigos fontes.

Todas essas ferramentas possuem licença gratuita, código fonte aberto, desenvolvidas com a linguagem MathScript e possuem recursos para trabalhar com

ferramentas de desenvolvimento como no caso proposto por esse trabalho o MATLAB 2015 *Student Version*.

Outra questão é que nos tutoriais e teses que se utilizam das ferramentas podem ser encontradas informações para sua instalação, além de links para códigos de terceiros que podem ser instalados e utilizados.

Com todas as essas características favoráveis apresentadas, os planos de trabalho simulado utilizados nesses trabalhos foram construídos com o auxílio da ferramenta MATLAB e os Toolboxes REBEL, EKF/UKF e Dynaest, mais informações podem ser encontradas em Arasaratnam e Haykin (2008), Bar-Shalom, Li e Kirubarajan (2001), Bishop (2006), Corke (2011), Hartikainen e Särkkä (2008), Haykin (2001), Julier e Uhlmann (1997), Lagadic (2013) e Särkkä, Vehtari e Lampinen (2007).

#### **4.4 Resultados Esperados**

Os principais resultados esperados nesta pesquisa são listados a seguir:

- Concepção de modelos de predição de sistemas de Servovisão sujeitos a ruídos e alteração de direção brusca e implementação desses modelos.
- Um estudo qualitativo sobre a semelhança dos sistemas geradores e dos gerados pelo modelo proposto e suas variantes.
- Um estudo qualitativo sobre algumas técnicas de predição utilizando Sistemas Estatísticos (Filtros de Kalman).
- Determinação dos algoritmos a serem utilizados no treinamento de Redes Neurais.
- Determinar a influência do posicionamento cartesianos dos Sensores de Rolamento sobre a acurácia da metodologia Bearing Only.

## 4.5 Delimitação do Escopo

O presente trabalho apresenta um estudo e avaliação de metodologias conhecidas para o tratamento de não linearidades em operações de Servovisão aplicado em objetos em movimento. Para isso uma simulação é realizada para concepção do Sistema de Servovisão e o avaliação do problema.

Embora na simulação todo o sistema de servovisão será considerado, nessa proposta são levados em consideração apenas as não linearidades do sistema, desconsiderando outros problemas comuns em Servovisão e outras abordagens adicionais como:

- Aquisição e extração de características da imagem;
- Calibração da câmera;
- Configurações da câmera;
- Reconstrução 3D;
- Detalhes sobre algoritmos adicionais utilizados ou que podem ser encontrados em outros trabalhos;
- Outros problemas que fazem parte do estudo de Servovisão;

O único evento inesperado que é considerado nesse trabalho é a ocorrência de *não linearidades nos movimentos*, demais eventos inesperados são desconsiderados e uma revisão sobre os demais tópicos podem ser encontradas em Corke (2011), Hutchinson, Hager e Corke (1996), Chaumette e Hutchinson (2006) e Chaumette e Hutchinson (2007), Bar-Shalom, Li e Kirubarajan (2001) e nos artigos relacionados, dispostos no Capítulo 3 (Revisão Bibliográfica).

# Capítulo 5

## IMPLEMENTAÇÃO

---

---

Conforme descrito anteriormente o objetivo desse trabalho é estudar a Estruturação Ambiental, sendo implementado métodos para predição de series, algoritmos de treinamento de redes neurais e a determinação de posição e velocidade de um objeto utilizando a metodologia (*Bearing Only*)

### 5.1 Algoritmo do Filtro EKF

Para compreender o algoritmo EKF, pode-se assumir um sistema dinâmico não-linear, podendo ser descrito por um modelo de espaço de estados, onde  $W_k$  e  $V_k$  encontrados nas Equações 5.1 e 5.2 são processos de ruído branco gaussiano com média zero independente das matrizes de covariância  $R_k$  e  $Q_k$ .

A função  $f(k, X_k)$  é uma matriz de transição de uma função não linear, provavelmente sofrendo variação ao longo do tempo. A função  $h(k, X_k)$  é uma matriz composta por medidas não lineares, podendo ser variantes no tempo (HAYKIN, 2001).

$$X_{k+1} = f(k, X_k) + W_k, \quad (5.1)$$

$$Y_k = h(k, X_k) + V_k, \quad (5.2)$$

A ideia básica do EKF de acordo com Haykin (2001) é linearizar o modelo de espaço de estados, sendo definido pelas equações 5.3 e 5.4, de modo que para cada instância em função do tempo, os valores atuais estão mais próximos do estimado para cada atualização do estado.

Essa atualização é realizada por  $\hat{\mathbf{X}}_k$  ou  $\hat{\mathbf{X}}_k^-$ , a função depende particularmente ao problema considerado.

Uma vez que a representação do modelo linear é obtida a partir da aplicação das equações do EKF. Sendo possível dizer que a aproximação realizada pelo filtro EKF, pode ser descrita em duas fases distintas: Na primeira fase a construção das matrizes descritas nas Equações 5.3 e 5.4;

$$\mathbf{F}_{k+1,k} = \left. \frac{\partial f(k, \mathbf{X})}{\partial \mathbf{X}} \right|_{\mathbf{X} = \hat{\mathbf{X}}_k} \quad (5.3)$$

$$\mathbf{H}_k = \left. \frac{\partial h(k, \mathbf{X}_k)}{\partial \mathbf{X}} \right|_{\mathbf{X} = \hat{\mathbf{X}}_k^-} \quad (5.4)$$

Nesta fase, a entrada ij-ésimo  $\mathbf{F}_{k+1,k}$  é idêntico ao da derivada parcial do componente i-ésimo  $\mathbf{F}_{k,X}$  em relação ao componente de ordem j-ésimo de  $\mathbf{X}$ .

De igual modo, a entrada ij-ésimo  $\mathbf{H}_k$  é idêntica à parte  $i$  da derivada parcial de  $H(k, \mathbf{X})$  em relação ao componente de ordem j-ésimo de  $\mathbf{X}$ .

No primeiro caso, as derivadas de  $\hat{\mathbf{X}}_k$  são avaliadas, de modo que no último caso, as derivadas serão avaliadas em  $\hat{\mathbf{X}}_k^-$ . As entradas das matrizes  $\mathbf{F}_{k+1,k}$  e  $\mathbf{H}_k$  são ambas conhecidas como sendo calculadas por  $\hat{\mathbf{X}}_k$  e  $\hat{\mathbf{X}}_k^-$ , sendo avaliados no momento  $k$  (HAYKIN, 2001).

Na segunda fase do Filtro EKF, a matriz  $\mathbf{F}_{k+1,k}$  e  $\mathbf{H}_k$  são avaliadas na sequência em que são utilizados na função de primeira ordem de aproximação Taylor para realizar funções não lineares  $\mathbf{F}(k, \mathbf{X}_k)$  e  $\mathbf{H}(k, \mathbf{X}_k)$  rotacionada  $\hat{\mathbf{X}}_k$  e  $\hat{\mathbf{X}}_k^-$ , respectivamente. De modo específico  $\mathbf{F}(k, \mathbf{X}_k)$  e  $\mathbf{H}(k, \mathbf{X}_k)$  são estimadas da forma como segue:

$$\mathbf{F}(k, \mathbf{X}_k) \approx \mathbf{F}(\mathbf{X}, \hat{\mathbf{X}}_k) + \mathbf{F}_{k+1,k}(\mathbf{X}, \hat{\mathbf{X}}_k) \quad (5.5)$$

$$\mathbf{H}(k, \mathbf{X}_k) \approx \mathbf{H}(\mathbf{X}, \hat{\mathbf{X}}_k^-) + \mathbf{H}_{k+1,k}(\mathbf{X}, \hat{\mathbf{X}}_k^-) \quad (5.6)$$

Com a estimativa das expressões 5 e 6, podemos realizar o processo de aproximação em equações não-lineares 1 e 2, como mostrado abaixo:

$$\mathbf{X}_{k+1} \approx \mathbf{F}_{k+1,k} \mathbf{X}_k + \mathbf{W}_k + \mathbf{d}_k \quad (5.7)$$

$$\bar{\mathbf{Y}}_k \approx \mathbf{H}_k \mathbf{X}_k + \mathbf{V}_k \quad (5.8)$$

Esta parte introduz duas novas quantidades a serem calculadas:

$$\bar{\mathbf{Y}}_k = \mathbf{Y}_k - \{h(\mathbf{X}, \hat{\mathbf{X}}_k^-) - \mathbf{H}_k \hat{\mathbf{X}}_k^-\} \quad (5.9)$$

$$\mathbf{d}_k = \mathbf{f}(\mathbf{X}, \hat{\mathbf{X}}_k) - \mathbf{F}_{k+1,k} \hat{\mathbf{X}}_k \quad (5.10)$$

As entradas em  $\bar{\mathbf{Y}}_k$  são todos conhecidos no momento  $k$ , portanto  $\bar{\mathbf{Y}}_k$  pode ser considerado como um vetor de observação no tempo  $n$ . Da mesma forma, as entradas no termo  $\mathbf{d}_k$ , são todas conhecidas no momento  $k$  (Haykin, 2001).

## 5.2 Treinamento utilizando o filtro EKF

No Treinamento de uma Rede Neural utilizando o Filtro EKF, uma MLP é utilizada, como mostrado na Figura 5-1. Esta rede é composta por uma camada de



entrada, uma camada oculta e uma camada de saída. Os valores de entrada têm sua respectiva saída, a partir dos pesos e da função de mapeamento  $\mathbf{G}$ .

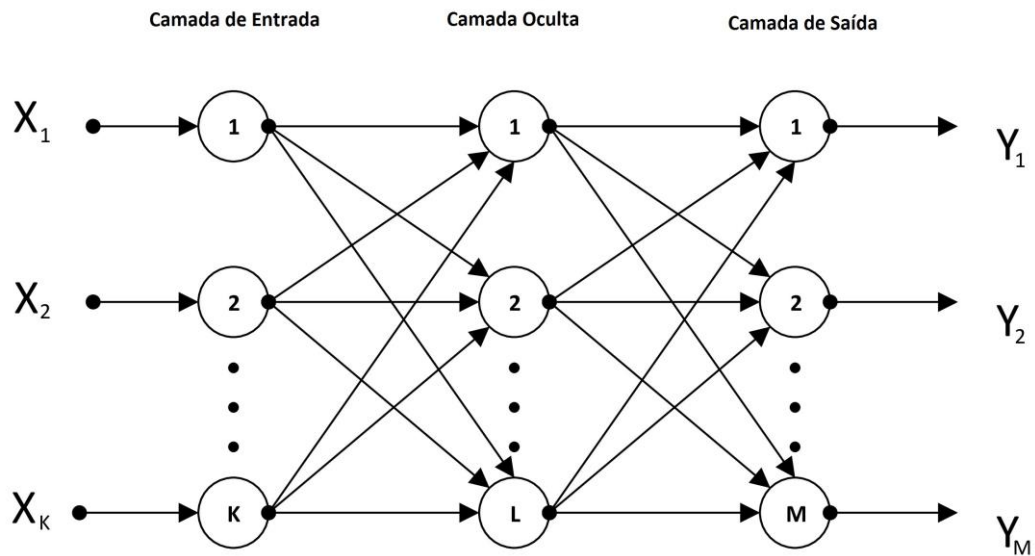


Figura 5-1 - Estrutura de uma Rede Neural Multicamadas.

Para realizar o treinamento de uma rede neural em primeiro lugar é necessário organizar todas as entradas, saídas e os pesos de rede como vetores de estado (ZHAN e WAN, 2006) (HARTIKAINEN e SÄRKKÄ, 2008).

Em seguida, a rede calcula o EKF e as mesmas ponderações são utilizadas para modificar as previsões de estimativa de vetores de estado do filtro, portanto as observações são processadas. Para descrever o treinamento da rede, pode-se supor um problema de estimação de estados onde você tem as seguintes Equações 5.11 e 5.12, de dinâmica e observações descritas em IIGUNI (1992).

$$W_k = W_{k-1} \tag{5.11}$$

$$dk = Y_k + V_k = g(W_k, X_k) + V_k \tag{5.12}$$

Where  $X = [x_1, x_2, \dots, x_k]^T$ ,  $Y = [y_1, y_2, \dots, y_m]^T$ ,  $W = [(W_1)^T, (W_2)^T]^T$ ,  $W_1 = [(W_{11})^T, (W_{12})^T, \dots, (W_{1L})^T]^T$ ,  $W_2 = [(W_{21})^T, (W_{22})^T, \dots, (W_{2M})^T]^T$ ,  $W_{1l} = [W_{1l,1}, W_{1l,2}, \dots, W_{1l,k}]^T$ ,  $W_{2m} = [W_{2m,1}, W_{2m,2}, \dots, W_{2m,L}]^T$ , and  $1 \leq m \leq M, 1 \leq l \leq L$  (ZHAN e WAN, 2006).

As variáveis  $d$  e  $y$  são as saídas desejadas,  $v$  é o ruído de observação aleatória. Vamos assumir um Ruído Branco Gaussiano com média  $R$  da matriz de covariância.

Por causa de uma série de artigos que descrevem o EKF, o mesmo será representado de forma sucinta a seguir:

$$S_{k+1} = H_{k+1} P_k H_{k+1}^T + R_{k+1} \quad (5.13)$$

$$K_{k+1} = P_k H_{k+1}^T S_{k+1}^{-1} \quad (5.14)$$

$$\widehat{W}_{k+1} = \widehat{W}_k + K_{k+1} [dk - g(\widehat{W}_k, X_k)] \quad (5.15)$$

$$P_{k+1} = P_k - K_{k+1} H_{k+1} P_k \quad (5.16)$$

Deste modo  $K_{k+1}$  é o ganho do filtro de Kalman,  $P_k$  é a matriz de covariância de erro de aproximação, e  $H_{k+1}$  é a matriz Jacobiana da função  $G$  em relação ao estado  $W$  na estimativa corrente  $\widehat{W}_k$ .

### 5.3 Algoritmo do Filtro UKF

O algoritmo UKF utilizando a Unscented Transformation (UT) é uma maneira de calcular as estatísticas de uma variável aleatória, que permite uma transformação não-linear (JULIER e UHLMANN, 1997).

Considere-se a propagação de uma variável  $X$  ( tamanho  $n$  ) através de uma função não linear  $Y = f ( X )$  . Assumindo que  $X$  é significante em  $\bar{X}$  e na covariância  $P_X$  . Para calcular as estatísticas  $Y$ , na forma de uma matriz composta por  $\chi$  até  $2L + 1$  vetores sigma  $\chi^i$  como se segue:

$$\begin{aligned}\chi^0 &= \bar{X}, \\ \chi^i &= + (p (L + \lambda)PX)^i, i = 1, \dots, L, \\ \chi^i &= \bar{X} - (p (L + \lambda)PX)^{i-L}, i = L + 1, \dots, 2L,\end{aligned}\tag{5.17}$$

Onde  $\lambda \in \mathbb{R}$  (Reais),  $(p (L + \lambda)PX)^i$  é o  $i$ -ésima linha ou coluna da matriz raiz quadrada de  $(L + \lambda)PX$  e  $W_i$  é o peso que esta associada com o  $i$ -ésimo ponto.

O procedimento de transformação é o seguinte Julier e Uhlmann (1997): O primeiro passo é instanciar cada ponto por meio da função para se obter o conjunto de pontos sigma transformados,  $Y_i = f [ X_i ]$ .

No segundo passo, a média é dada pela média ponderada dos pontos transformados:

$$\bar{Y} = \sum_{i=0}^{2L} W_i Y_i.\tag{5.18}$$

No terceiro passo, a covariância são os pesos correspondentes a rede neural, sendo este outro produto dos pontos transformados:

$$\mathbf{P}_X = \sum_{i=0}^{2L} W_i \{Y_i - \bar{Y}\} \{Y_i - \bar{Y}\}^T . \quad (5.19)$$

Após aprender os conceitos da execução da UT, na aplicação do Filtro de Kalman durante essas etapas: Para prever um novo estado do sistema  $\bar{X}(\lambda + 1 | \lambda)$  e sua covariância associada a  $\mathbf{P}(\lambda + 1 | \lambda)$ . Para realizar essa previsão é necessário compreender os efeitos do ruído no processo. Predizendo a observação esperada  $\hat{z}(\lambda + 1 | \lambda)$  e a covariância da inovação  $\mathbf{P}_{vv}(\lambda + 1 | \lambda)$ , esta predição deve incluir os efeitos do ruído de observação.

Finalmente, prevendo a matriz de correlação cruzada  $\mathbf{P}_{xz}(\lambda + 1 | \lambda)$ .

Estes passos podem ser facilmente alterados, reconstruindo os vetores de processos e os modelos de observação de estados.

Para fazê-lo, o vetor de estados é estendido, com o processo em termos de níveis de ruído para um vetor dimensional  $L^a = L + q$ , em que :

$$\mathbf{X}^a(\lambda) = \begin{bmatrix} \mathbf{X}(\lambda) \\ \mathbf{V}(\lambda) \end{bmatrix} .$$

O modelo de processo pode ser reescrito em função de  $\mathbf{X}^a(\lambda)$  onde,  $\mathbf{X}(\lambda + 1) = \mathbf{f}[\mathbf{X}^a(\lambda), \mathbf{u}(\lambda), \lambda]$ . A transformação UT utiliza-se de  $2rta + 1$  pontos sigma, que podem ser descritos da seguinte maneira:

$$\hat{\mathbf{X}}^a(\lambda|\lambda) = \begin{pmatrix} \hat{\mathbf{X}}(\lambda|\lambda) \\ \mathbf{0}_{(q)x(1)} \end{pmatrix} e$$

$$\mathbf{P}^a(\lambda|\lambda) = \begin{bmatrix} \mathbf{P}(\lambda|\lambda) & \mathbf{P}_{xv}(\lambda|\lambda) \\ \mathbf{P}_{xv}(\lambda|\lambda) & \mathbf{Q}(\lambda) \end{bmatrix}. \quad (5.20)$$

As matrizes no com domínio diagonal são as covariâncias e as não diagonais são sub-blocos das correlações entre os erros de estado e os ruídos de processo.

Embora este método necessite do uso de pontos sigma adicionais, isto significa que os efeitos do ruído do processo (em termos de impacto sobre a média e covariância) são introduzidos com a mesma ordem de precisão como a incerteza no estado (JULIER e UHLMANN, 1997).

## 5.4 Treinamento utilizando o Filtro UKF

O treinamento de uma RN utilizando o algoritmo UKF para a formação de acordo com Zhan e Wan (2006) é muito semelhante ao EKF. Da mesma maneira que no EKF, os pesos das ligações devem estar dispostos no formato de vetor de estados.

Para o UKF os estados são calculados pelo Unscented Transform ( UT) . Assim, se propagação é realizada de forma analítica do sistema não linear, sem a necessidade de realizar a medição da matriz Jacobiana.

O UKF é apresentado a seguir em quatro etapas básicas descritas em Zhan e Wan (2006):

- Etapa primeira (Inicialização):

$$\widehat{W}0 = E[W0] \quad (5.21)$$

$$P0 = E[(W0 - \widehat{W}0)(W0 - \widehat{W}0)^T]. \quad (5.22)$$

- Segunda Etapa (Cálculo dos Pontos Sigma):

$$\left\{ \begin{array}{l} \chi_{0,k-1} = \widehat{W}_k - 1 \quad \omega_0 = \frac{k}{(L+k)} \\ \chi_{i,k-1} = \widehat{W}_{k-1} + \sqrt{(p(L+k)P_{k-1})_i} \quad \omega_i = \frac{0.5}{(L+k)} \\ \chi_{i+L,k-1} = \widehat{W}_{k-1} - \sqrt{(p(L+k)P_{k-1})_i} \quad \omega_{i+L} = \frac{0.5}{(L+k)} \end{array} \right. \quad (5.23)$$

Onde  $i = 1, 2, \dots, L$  e  $L$  é a dimensão do estado. O parâmetro  $k$  é utilizado para controle da matriz de covariância.

- Terceira etapa (Atualização Temporal):

$$\chi_{i,k|k-1} = \chi_{i,k-1} \quad (5.24)$$

$$\widehat{W}_k^- = \sum_{i=0}^{2L} \omega_i \chi_{i,k|k-1} \quad (5.25)$$

$$P_k^- = \sum_{i=0}^{2L} \omega_i [\chi_{i,k|k-1} - \widehat{W}_k^-][\chi_{i,k|k-1} - \widehat{W}_k^-]^T \quad (5.26)$$

$$\gamma_{i,k|k-1} = g(\chi_{(i,k|k-1)}, X_k) \quad (5.27)$$

$$\hat{Y}_k^- = \sum_{i=0}^{2L} \omega_i \gamma_{i,k|k-1}. \quad (5.28)$$

- Etapa Quatro (Atualização das Medidas):

$$S_k = \sum_{i=0}^{2L} \omega_i [\gamma_{i,k|k-1} - \hat{Y}_k^-][\gamma_{i,k|k-1} - \hat{Y}_k^-]^T + R_k \quad (5.29)$$

$$G_k = \sum_{i=0}^{2L} \omega_i [\chi_{i,k|k-1} - \hat{W}_k^-][\gamma_{i,k|k-1} - \hat{Y}_k^-]^T \quad (5.30)$$

$$\hat{W}_k = \hat{W}_k^- + G_k S_k^{-1} (d_k - \hat{Y}_k^-) \quad (5.31)$$

$$P_k = P_k^- - \kappa_k S_k \kappa_k^T. \quad (5.32)$$

## 5.5 Modelo de Posicionamento usando sensores angulares (Bearing Only)

O estado do objeto na iteração  $k$  consiste na posição em coordenadas cartesianas bidimensional  $x_k$  e  $y_k$ , e a velocidade em direção a esses eixos de coordenadas,  $\dot{x}_k$  e  $\dot{y}_k$ .

Sendo assim, o vetor de estado pode ser expresso como sendo:

$$x_k = (x_k \ y_k \ \dot{x}_k \ \dot{y}_k)^T \quad (5.33)$$

A dinâmica do objeto a ser rastreado pode ser modelada de maneira linear, discretizado a partir da do Modelo de Velocidade de Wiener. (BAR-SHALOM, LI e KIRUBARAJAN, 2001) (HARTIKAINEN e SÄRKKÄ, 2008).

$$x_k = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{pmatrix} + q_{k-1}, \quad (5.34)$$

Segundo Hartikainen e Särkkä (2008) quando  $q_{k-1}$  é um ruído de processo gaussiano de média zero e covariância:

$$E[q_{k-1} q_{k-1}^T] = \begin{pmatrix} \frac{1}{3}\Delta t^3 & 0 & \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{3}\Delta t^3 & 0 & \frac{1}{2}\Delta t^2 \\ \frac{1}{2}\Delta t^2 & 0 & \Delta t & 0 \\ 0 & \frac{1}{2}\Delta t^2 & 0 & \Delta t \end{pmatrix} q, \quad (5.35)$$

De modo que  $q$  é a densidade espectral do ruído a qual é definida como  $q = 0,1$  nas simulações. O modelo de medição para o sensor  $i$  é definido como:

$$\theta_k^i = \arctan\left(\frac{y_k - s_y^i}{x_k - s_x^i}\right) + r_k^i \quad (5.36)$$

Sendo melhor descrito em Simo Särkkä (2004), quando a posição do sensor  $i$  e  $r_k^i \sim N(0, \sigma^2)$ , com  $\sigma = 0.05$  radianos.



As derivadas do modelo de medição, sendo estas necessárias à execução do EKF, podem ser descritas como sendo as Equações 5.37, 5.38 e 5.39:

$$\frac{\partial^2 \mathbf{h}^i(x_k)}{\partial x_k \partial x_k} = \frac{-2(x_k - s_x^i)}{((x_k - s_x^i)^2 + (y_k - s_y^i)^2)^2} \quad (5.37)$$

$$\frac{\partial^2 \mathbf{h}^i(x_k)}{\partial x_k \partial y_k} = \frac{(x_k - s_x^i)^2 + (y_k - s_y^i)^2}{((x_k - s_x^i)^2 + (y_k - s_y^i)^2)^2} \quad (5.38)$$

$$\frac{\partial^2 \mathbf{h}^i(x_k)}{\partial y_k \partial y_k} = \frac{-2(y_k - s_y^i)}{((x_k - s_x^i)^2 + (y_k - s_y^i)^2)^2} \quad (5.39)$$

O objeto tem início de deslocamento com o estado  $x_0 = (0 \ 0 \ 1 \ 0)$ , e na estimativa será definida a distribuição prévia para o estado conforme citado por Hartikainen e Särkkä (2008) e Simo Särkkä (2004) como sendo  $x_0 \sim N(\mathbf{0}, \mathbf{P}_0)$  portanto teremos na equação 5.40.

$$\mathbf{P}_0 = \begin{pmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{pmatrix} \quad (5.40)$$

# Capítulo 6

## RESULTADOS OBTIDOS

---

Com o propósito de demonstrar as características do método de treinamento de Redes Neurais e Estruturação Ambiental proposto nesse trabalho, esse capítulo descreve os experimentos e os testes realizados durante a pesquisa.

### 6.1 Avaliação das Modificações do Filtro de Kalman

A partir de uma composição de funções trigonométricas (principalmente seno e cosseno).

Um sistema de referência, apresentado em Kotecha e Djuric (2003) e Wu Hu, *et al.* (2006), como um desafio para a área de Estimação de Estados não lineares é dado pela Equação 6.1:

$$x_n = \alpha x_n - 1 + \beta \frac{x_{n-1}}{1+x_{n-1}^2} + \gamma \cos(1.2(n-1)) + u_n \quad (6.1)$$

Para  $\mu_n \sim N(0, \sigma_\mu^2)$  e  $\vartheta_n \sim N(0, \sigma_\mu^2)$ .

Nessa comparação preliminar são utilizados os seguintes parâmetros = 1,  $\sigma_v^2 = 1$ ,  $x_0 = 0.1$ ,  $\alpha = 0.5$ ,  $\beta = 25$ ,  $\gamma = 8$  e  $N=500$ . A inserção do termo cosseno na equação de transição de estado, simula o efeito de variáveis no ruído temporal.

Existe dificuldade em prever novos estados, a existência de ciclos muito curtos, aumenta a dificuldade proporcionalmente (AUGER, HILAIRET, *et al.*, 2013) (HSIAO, 2011) (ZHAN e WAN, 2006).

A previsão dos sistemas complexos é a justificativa deste trabalho inicial, pois diversos autores relataram a dificuldade de previsão de séries temporais, alguns deles são Auger, Hilairet, *et al.* (2013), Hsiao (2011), Zhan e Wan (2006), Kurt e Yavuz (2012), Chen (2012), Corke e Good (1993).

Para prever essas séries ruidosas é necessária uma metodologia recursiva eficiente. Modificações do filtro de Kalman são estudados para esta finalidade.

Este trabalho comparou várias dessas mudanças, afim de verificar qual é a com maior eficiência na detecção de novos estados em uma serie complexa de referência.

Sendo oito dessas metodologias testadas utilizando como ferramenta o EKF / UKF Toolbox (HARTIKAINEN e SÄRKKÄ, 2008):

Unscented Kalman Filter First Order (**UKF 1**)

Unscented Kalman Filter First Order Smoothed (**URTS 1**)

Unscented Kalman Filter Second Order (**UKF 2**)

Unscented Kalman Filter Second Order Smoothed (**URTS 2**)

Extended Kalman Filter (**EKF**)

Extended Kalman Filter Smoothed (**ERTS**)

Spherical-Radial Cubature Kalman Filter (**CKF**)

### Spherical-Radial Cubature Kalman Filter Smoothed (**CRTS**)

A denominação Kalman *Smother* (RTS) utilizada nesse trabalho, se refere a modificação conhecida como Rauch-Tung-Striebel *Smother* (H. E. RAUCH, STRIEBEL e TUNG, 1965).

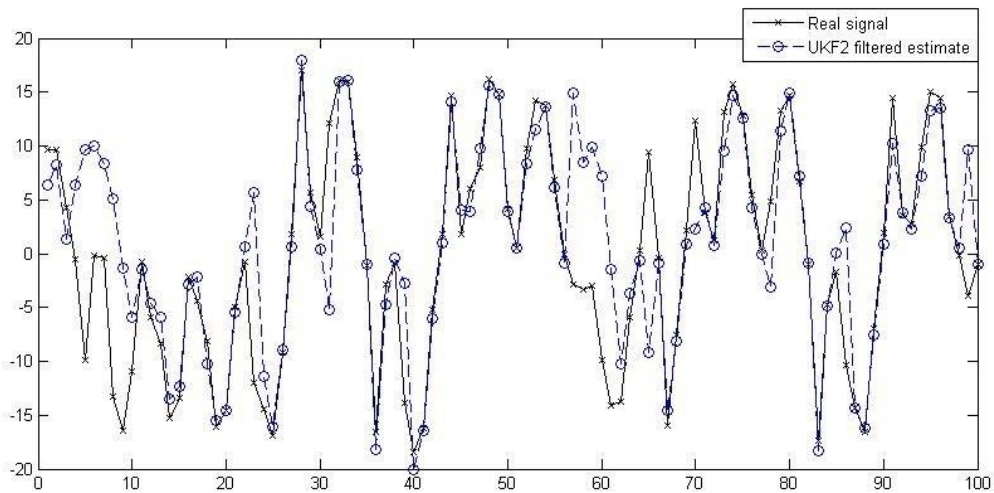
Segundo Hromkovič (2004) designa-se por método de Monte Carlo (MMC) qualquer método de uma classe de métodos estatísticos que se baseiam em amostragens aleatórias massivas para obter resultados numéricos, isto é, repetindo sucessivas simulações um elevado número de vezes, para calcular probabilidades heurísticamente, tal como se, de fato, se registrassem os resultados reais em jogos de casino. Este tipo de método é utilizado em simulações estocásticas com diversas aplicações em áreas como a física, matemática e biologia.

O método de Monte Carlo tem sido utilizado há bastante tempo como forma de obter aproximações numéricas de funções complexas em que não é viável, ou é mesmo impossível, obter uma solução analítica ou, pelo menos, determinística.

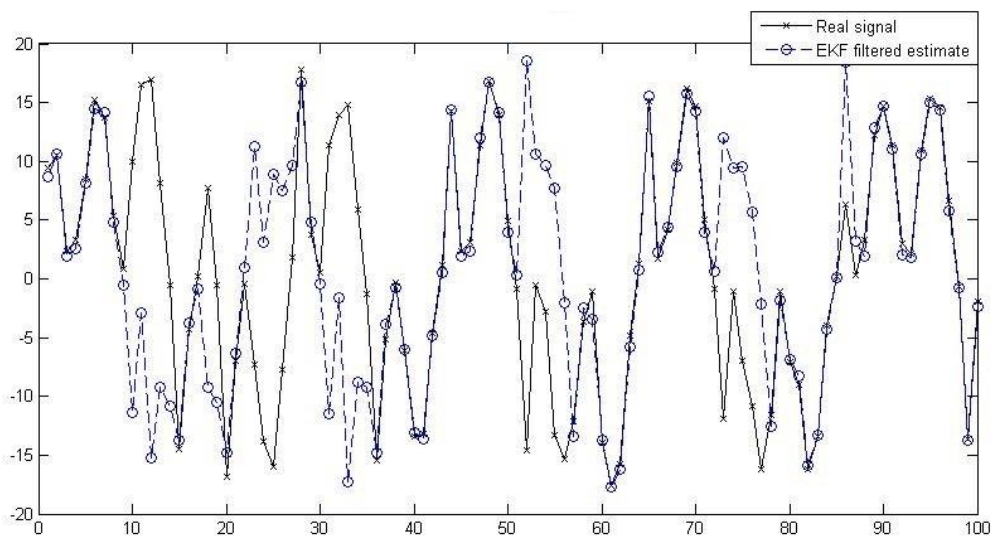
Os resultados de alguns dos métodos de 100 amostragens de Monte Carlo serão discutidos a seguir.

A comparação entre UKF 2 mostrado na Fig. 6-1 e filtro EKF mostrado na Fig. 6-2, evidencia um melhor desempenho do Filtro UKF 2.

Este resultado pode ser explicado pela sua eficácia no tratamento de cálculos e de linearização de matrizes, é necessária avaliação dos dois Filtros UKF, o de primeira e segunda ordem.

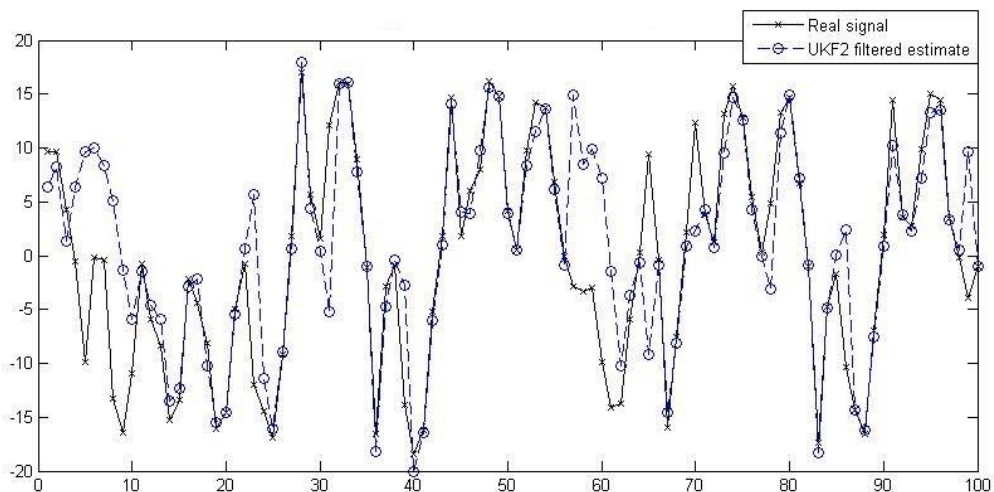


**Figura 6-1 - Unscented Kalman Filter de Segunda Ordem.**

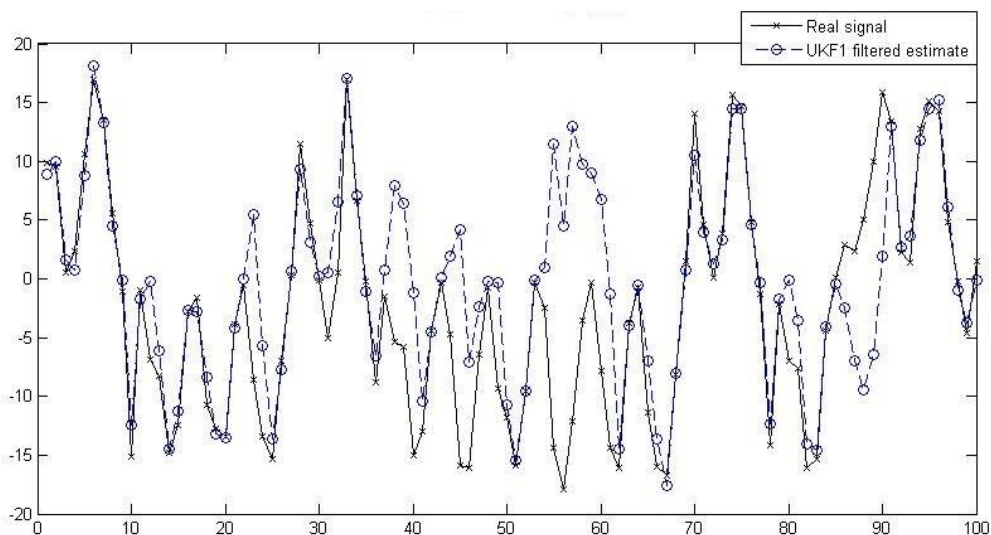


**Figura 6-2 - Extended Kalman Filter.**

Fig.6-3 e Fig. 6-4 avaliam o desempenho de Filtro UKF1 e UKF2 na predição de novos estados em sistemas não-lineares, tentando assim encontrar o método que resolva de melhor forma o sistema proposto, o Filtro UKF2 demonstra sua precisão e melhores resultados em comparação ao Filtro UKF1.

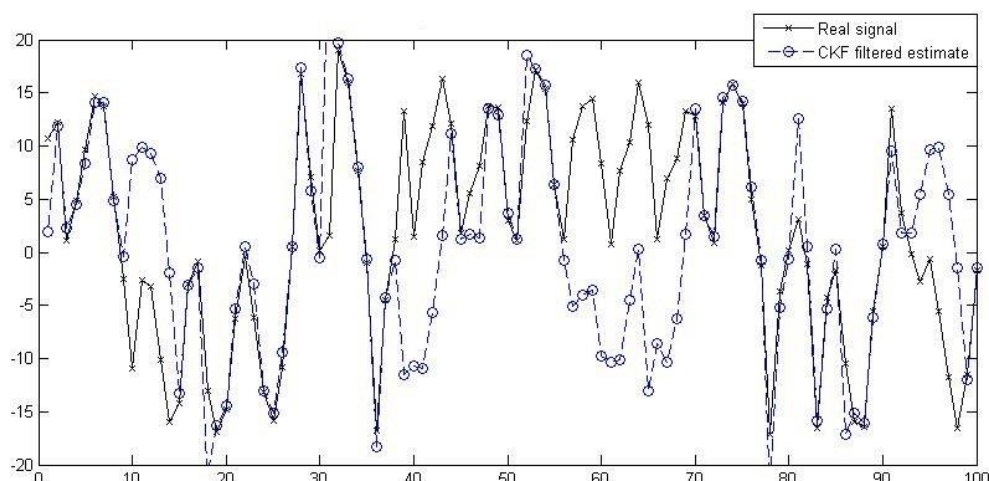


**Figura 6-3 - Unscented Kalman Filter de Segunda Ordem.**



**Figura 6-4 - Unscented Kalman Filter de Primeira Ordem.**

A Figura 6-5 mostra o resultado obtido com o Filtro CKF, as interações entre os estágios 35 e 45 mostram grande discrepância do valor obtido pelo filtro durante a simulação.



**Figura 6-5 - Cubature Kalman Filter.**

A Tabela 6-1, evidenciou a eficácia do Filtro de Segunda Ordem UKF, comparação com os outros filtros, assim a proposta para encontrar uma metodologia eficaz para posterior treinamento de redes neurais e localização espacial em ambientes não estruturados, tentando prever novos estados em sistemas de posicionamento em ambientes não estruturados foi válida.

Filtros utilizando o Rauch-Tung-Striebel Smother obtiveram maior acurácia quando comparado aos filtros sem suavização.

Os resultados gráficos das modificações utilizando suavizador, foram emitidas para evidenciar o desempenho dos filtros EKF e UKF de primeira e segunda ordem.

Em nosso trabalho de pesquisa os algoritmos a serem utilizados nas próximas análises serão as modificações do Filtro de Kalman EKF e UKF de primeira e segunda ordem e suas formas suavizadas, em função dos resultados obtidos conforme descritos na Tabela 6-1.

**Tabela 6-1** Comparação sobre as diferentes modificações do Filtro de Kalman.

Modificações	Erro Médio
	Quadrático (MSE)
	<i>Erro</i>
UKF 1	45.2114
URTS 1	41.8783
UKF 2	29.7576
URTS 2	25.3343
EKF	77.3210
ERTS	65.6581
CKF	74.5525
CRTS	73.7685

## 6.2 Resultados Treinamento das RN's

As simulações foram realizadas em na plataforma de desenvolvimento Matlab 2015 Student Version, utilizando a Biblioteca Recursive Bayesian Estimation Library (REBEL) desenvolvida por Wan e Van Der Merwe (2000) em um PC com processador AMD Quadricore 3.0- CPU GHz e 8 GB de memória.

Uma Rede Neural Multicamadas foi utilizada para aproximar a seguinte função não-linear:

$$y = x + 2 \cos(x)(-5)\sin(x) + x^3 \quad (6.2)$$

A entrada e saída  $\{ \tilde{x}_k, \tilde{y}_k \}$  da equação 6.2 estão livres de ruídos, sendo utilizados para gerar dados ruidosos utilizados no treinamento das RN .



Especificamente, 100 conjuntos de dados ruidosos  $\{\tilde{x}_k, \tilde{y}_k\}$  foram gerados aleatoriamente para treinamento, com  $\tilde{x}_k = x_k + \alpha v_k$ ,  $x_k = x \sim N(0, 1)$  e  $\tilde{y}_k = y_k + \beta \eta_k$ , onde  $k$  é o conjunto de dados do índice,  $k \in [1, 2, \dots, 100]$ , bem como os termos ruidosos  $v_k, \eta_k \sim N(0, 1)$ .

Os parâmetros  $\alpha$  e  $\beta$  são usados para controlar a intensidade do ruído. Foi utilizada como função de transferência da RN, a função Log-Sigmóide na camada oculta.

Sendo realizado o teste usando diversos números de neurônios (6, 8 e 10) na camada oculta, avaliando o impacto do número de neurônios na camada oculta, em função do Erro Quadrático Médio (MSE) obtido durante o treinamento da Rede Neural.

Na Tabela 6-2, o treinamento foi efetuado utilizando os seguintes parâmetros  $\alpha$  e  $\beta = 10^{-2}$ , que são responsáveis pela magnitude do grau do erro nos dados de entrada.

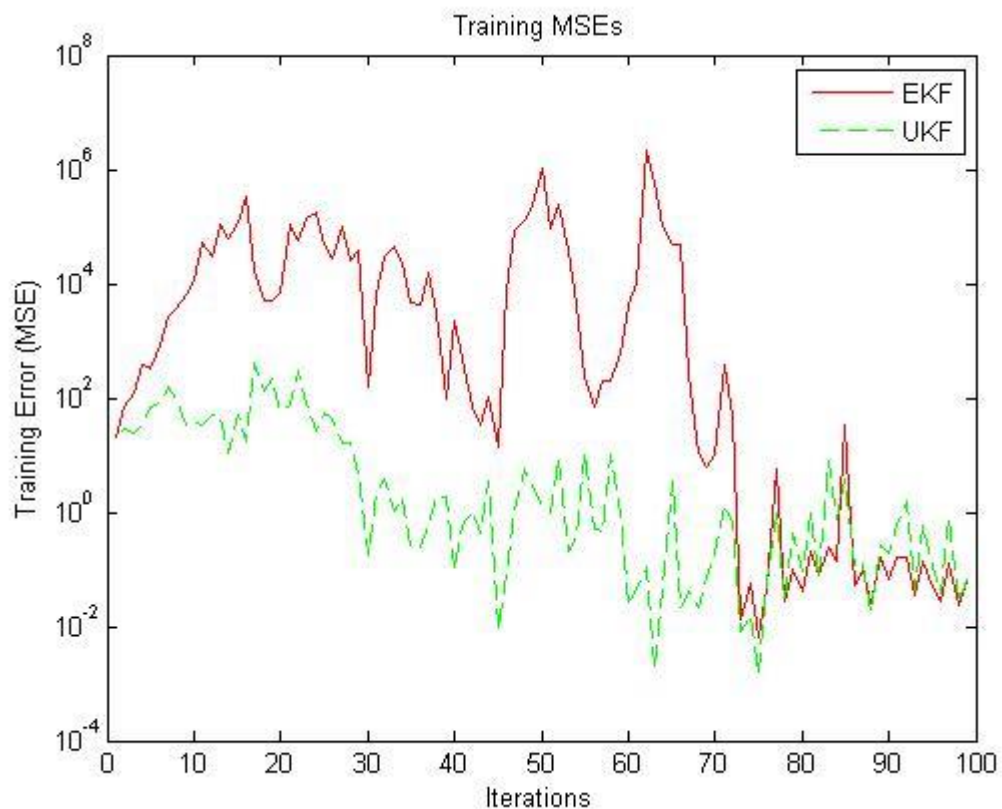
**Tabela 6-2 - Treinamento / Número de Neurônios**

Neurônios	MSE		Tempo (s)		R	
	UKF	EKF	UKF	EKF	UKF	EKF
<b>6</b>	5,48E <sup>-02</sup>	7,58E <sup>-02</sup>	0,53219	0,4000	0,9997	0,9994
<b>8</b>	<b>2,82E<sup>-02</sup></b>	<b>6,34E<sup>-03</sup></b>	<b>0,70524</b>	<b>0,5162</b>	<b>0,9994</b>	<b>0,9995</b>
<b>10</b>	6,70 <sup>-02</sup>	6,54E <sup>-02</sup>	0,8997	0,6787	0,9981	0,9992

Nota-se que utilizando 6 ou 10 os neurônios na camada oculta as ocorrências de erros aumentam durante o tempo de treinamento.

A Regressão Linear (R) foi avaliada em função dos dados de treinamento e resultados obtidos pela rede neural, indica a garantia de precisão do treinamento sendo utilizado como um parâmetro para determinar o número de neurônios ideal na camada oculta para um treinamento eficaz e com bom desempenho.

Foram utilizados 8 neurônios na camada oculta nos algoritmos de treinamento EKF e UKF, os resultados comparativos são mostrados na Figura 6-6.



**Figura 6-6 – Treinamento de RN utilizando o EKF e UKF ( $\alpha$  and  $\beta = 10^{-2}$ ).**

As comparações acima descritas foram realizadas do mesmo modo para os dados contidos na Tabela 6-2 utilizando (6 , 8 e 10) neurônios na camada oculta, utilizando os parâmetros  $\alpha$  e  $\beta = 10^{-3}$ .

Nota-se que o padrão de erros mostrados na Tabela 6-2 e Tabela 6-3 são repetidos quando 6 e 10 neurônios são utilizados na camada oculta, no entanto o filtro EKF utilizando 6 neurônios na camada oculta, obtém uma melhoria significativa em relação a utilização de 8 e 10 neurônios na camada oculta.

Porém optou-se por utilizar 8 neurônios na camada oculta, como meio comparador entre os algoritmos de treinamento EKF e UKF apresentados na Figura 6-7, porque a precisão é melhorada como demonstrado pelo parâmetro de regressão linear (R) na Tabela 6-3.

**Tabela 6-3 Treinamento / Número de Neurônios 2**

Neurônios	MSE		Time (s)		R	
	UKF	EKF	UKF	EKF	UKF	EKF
6	8,31E <sup>-02</sup>	1,01E <sup>-02</sup>	0,54224	0,39183	0,9999	0,9998
<b>8</b>	<b>3,97E<sup>-02</sup></b>	<b>1,54E<sup>-02</sup></b>	0,72356	0,51939	0,9996	0,9999
10	9,02E <sup>-02</sup>	2,44E <sup>-02</sup>	0,92541	0,62826	0,9997	0,9995

A Figura 6-7 demonstra a melhor taxa de aprendizagem Filtro UKF até iteração 75. Nas iterações 45, 63 e 75, há uma diminuição significativa no erro obtido durante o treinamento.

O EKF tende a obter um desempenho inferior na taxa de aprendizagem em comparação ao Filtro UKF até a iteração 70. As oscilações apresentadas na taxa de aprendizado, são devido à grande não-linearidades dos dados de entrada de acordo com a equação 6.2 utilizada na simulação.

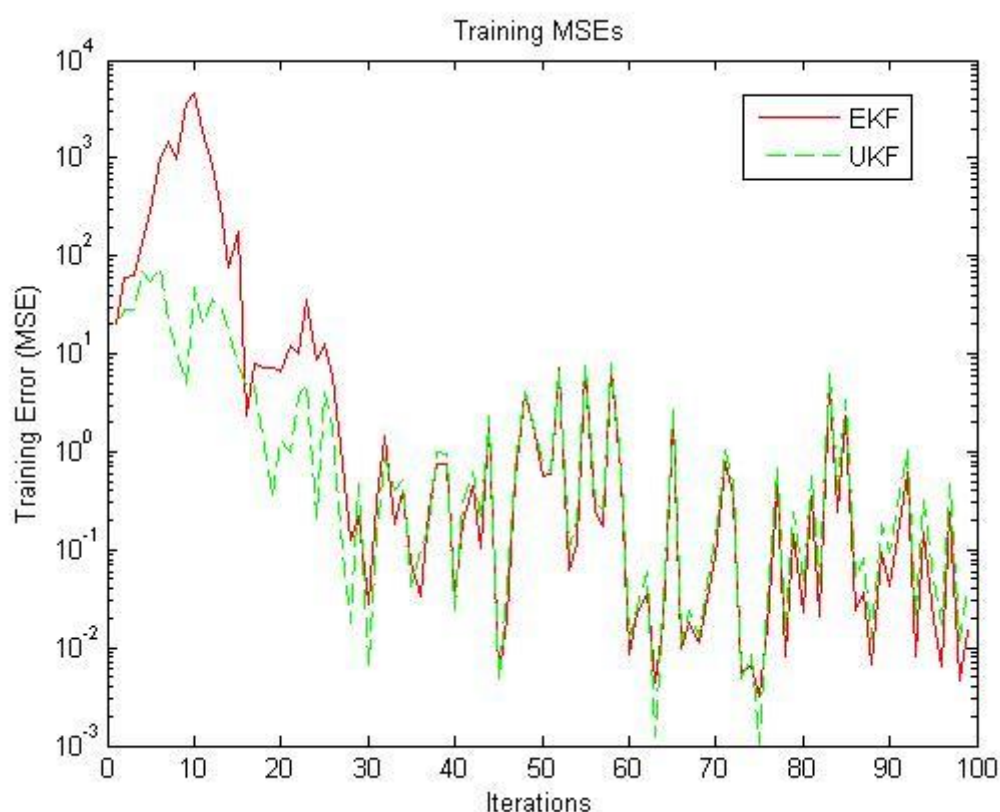


Figura 6-7 -Treinamento de RN utilizando o EKF e UKF ( $\alpha$  and  $\beta = 10^{-3}$ ).

A Figura 6-7 evidencia o filtro UKF mais rápido aprendizado até iteração 30. A partir da iteração 35, os dois Filtros de treinamento tendem a ter uma boa generalização da função, e o erro seguem próximos até a iteração 100.

Fica claro que um determinado dado de entrada com ruído de baixa ordem, as taxas de aprendizado do algoritmo de treinamento tendem a aumentar.

Os menores erros foram obtidos pelo filtro UKF nas iterações 45, 63 e 75, confirmando o melhor desempenho do filtro de UKF no treinamento de Redes Neurais para a predição em sistemas não lineares.

Portanto nota-se que o treinamento de redes neurais utilizando Filtros de Kalman é viável, como citado por Arasaratnam e Haykin (2008), Haykin (1999), Haykin (2001), Jeong-Gwan Kang (2010), Stubberud e Owen (1998).

Sendo necessário esforços para avaliar sua viabilidade no posicionamento em ambientes não estruturados, sendo objetivado nesse trabalho a comparação dos métodos em função do treinamento.

### 6.3 Resultados Estruturação Ambiental (Bearing Only)

O objetivo desses experimentos é proporcionar dados que permitam uma análise um pouco mais próxima das condições reais de sistema de rastreamento de objetos por Servovisão, enquanto que no primeiro tipo de experimento o objeto é avaliar a capacidade de previsão do objeto também em diferentes tipos de deslocamento.

Para as demonstrações de resultados das simulações foram utilizados o RMSE em 500 iterações de Monte Carlo.

Nos três tipos de experimentos o objeto é rastreado de forma planar (eixos  $x$  e  $y$ ), sendo a predição de dados referentes a posição e velocidade realizadas por dois sensores com retorno angular.

As três formas deslocamento utilizando o MathScript são descritas a seguir:

- Deslocamento linear com curvas;

$$a(1,50:150) = \pi/2/51/dt + 0.01*\text{randn}(1,101);$$
$$a(1,300:400) = -\pi/2/51/dt + 0.01*\text{randn}(1,101);$$

- Deslocamento Circular;

$$a(1,1:500) = \pi/2/100/dt + 0.01*\text{randn}(1,500);$$

- Deslocamento com variações bruscas de direção.

$$a(1,50:55) = \pi/2/4/dt + 0.01*\text{randn}(1,6);$$

$$a(1,150:155) = -\pi/2/5/dt + 0.01*\text{randn}(1,6);$$

$$a(1,250:255) = \pi/2/4/dt + 0.01*\text{randn}(1,6);$$

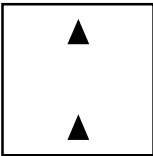
$$a(1,400:405) = \pi/2/5/dt + 0.01*\text{randn}(1,6);$$

Posicionamento dos sensores:

- **Posição 1**

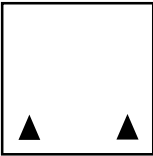
$S1 = (0; -2).$

$S2 = (0; 1).$


- **Posição 2**

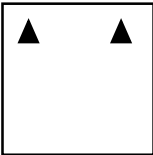
$S1 = (-1; -2).$

$S2 = (1; -2).$


- **Posição 3**

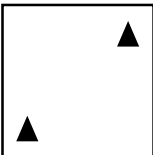
$S1 = (-1; 1).$

$S2 = (1; 1).$


- **Posição 4**

$S1 = (-1; -2).$

$S2 = (1; 1).$



Esses três tipos de movimentos permitem analisar o método estudado nesse trabalho em condições próximas a movimentação real, o rastreamento circular permite analisar situações de curvas e o deslocamento com variações bruscas permite verificar alterações drásticas na movimentação do objeto.

Detalhes sobre o método estudado nesse trabalho e utilizado junto a esses simuladores são discutidos nas sessões seguintes.

Na fase para as simulações, será utilizada uma metodologia clássica de filtragem, que nesse trabalho será utilizada para estruturação ambiental, sendo melhor descrita em Bar-Shalom, Li e Kirubarajan (2001), que tem como objetivo determinar a velocidade e posição de um objeto com sensores, que medem apenas os rolamentos (ou ângulos) do objeto.

Há um objeto em movimento na simulação e dois sensores angulares para determinar o seu percurso. Resolver este problema é importante, porque vários problemas de rastreamento e posicionamento, muitas vezes de caráter mais geral pode ser dividido em subproblemas, em que as metas individuais são controladas separadamente (SÄRKKÄ, VEHTARI e LAMPINEN, 2007).

### 6.3.1 Resultados (Deslocamento linear com curvas)

- **Trajectoria 1 Posição 1**

Na Figura 6-8 é demonstrada a trajetória real utilizada na estimativa do método, estando representada a trajetória e posicionamento dos sensores, referente as simulações que se seguem.

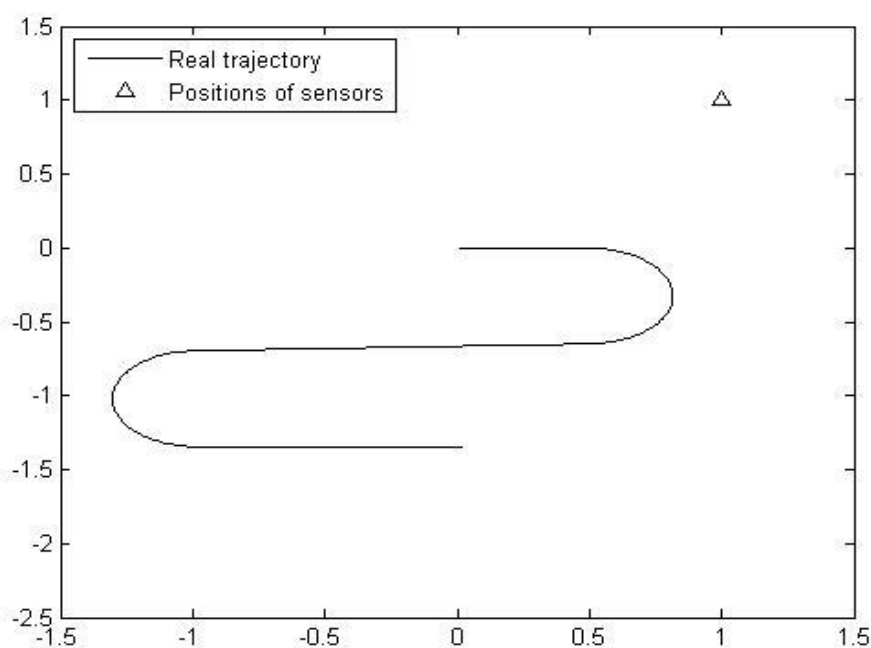
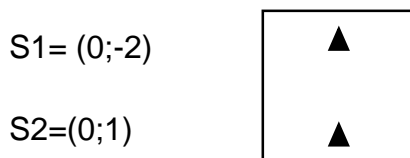


Figura 6-8 - Trajeto com curvas e sensores.

As simulações descritas na Figura 6-9, tem como principal característica o posicionamento dos sensores, e sua provável influência sobre o método proposto e os sistemas de filtragem.

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:



Nota-se durante a execução do algoritmo EKF na Figura 6-9, a dificuldade em determinar a velocidade durante as curvas efetuadas na primeira trajetória proposta para estudo.

Com a utilização do suavizador durante a execução, pode-se garantir uma boa generalização da predição sobre a posição do objeto no plano, porém as informações referente ao objeto em relação a velocidade demonstram uma menor efetividade do método.

A posição dos sensores pode se tornar algo crucial para melhora durante a predição, caso este que será estudado durante as simulações.

Durante a execução do algoritmo EKF de segunda ordem na Figura 6-10, pode-se notar uma melhor adaptabilidade do algoritmo as curvas da trajetória, e uma melhor predição referente a posição corrente do objeto na trajetória.



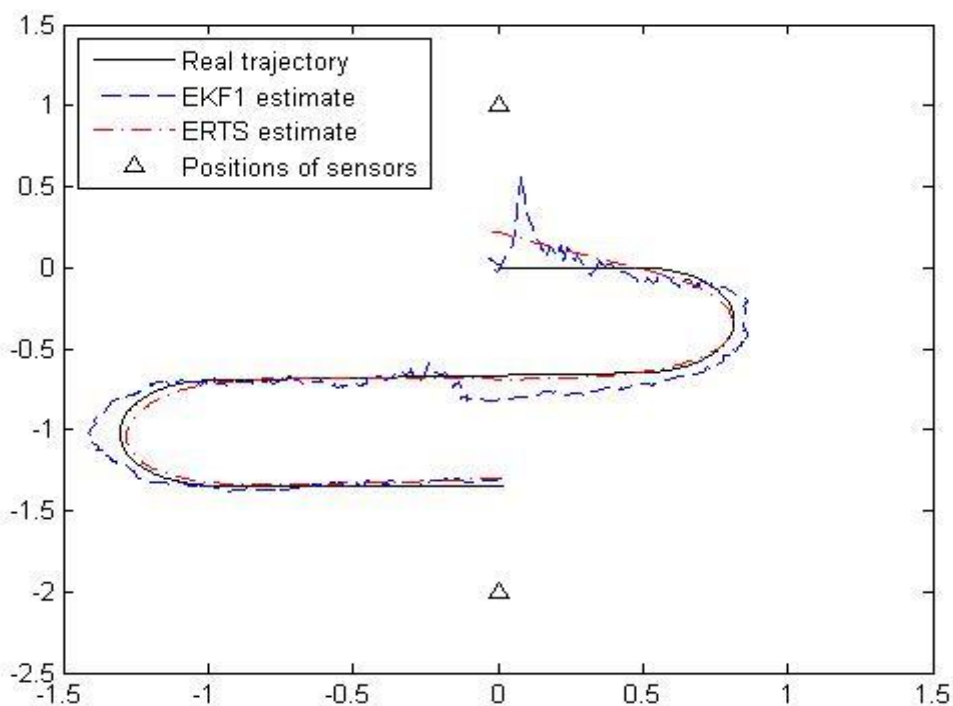


Figura 6-9 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.1).

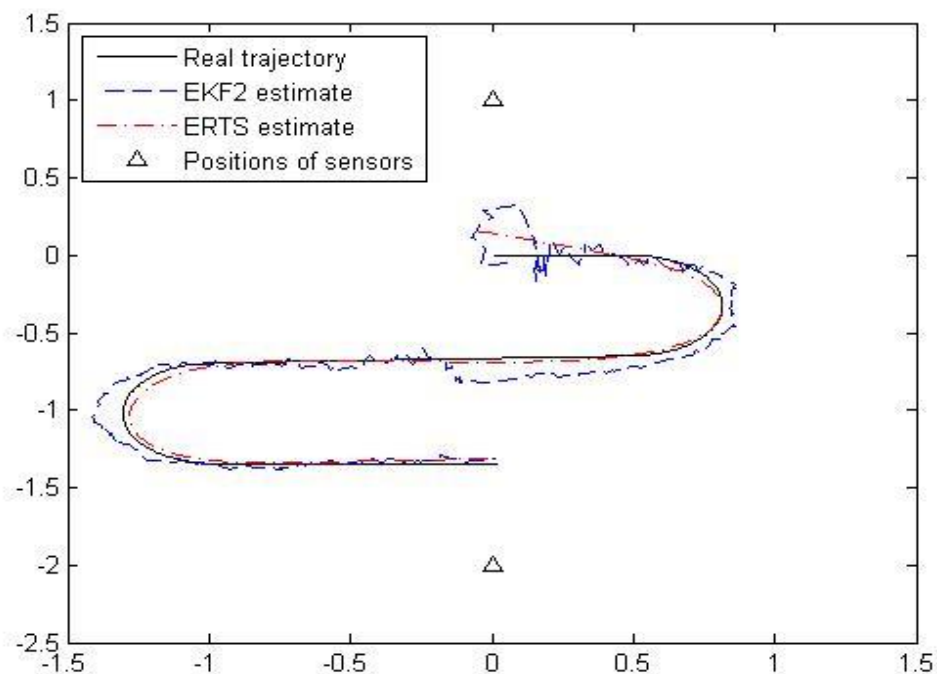


Figura 6-10 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.1).

A execução do algoritmo sobre curvas no trajeto proposto, nota-se uma menor deriva do algoritmo em relação ao algoritmo EKF de primeira ordem, sendo este conseguido pela boa generalização e melhor previsão de Filtros de ordem mais alta.

Com suavização realizada pelo algoritmo, verifica-se uma previsão bem próxima a trajetória proposta no estudo, ficando evidente o menor desvio em relação as medições quando realizada curvas durante a trajetória proposta.

A execução do Filtro UKF (Figura 6-11) mesmo sem a suavização, demonstra uma menor estabilidade da previsão em relação a trajetória proposta, essa situação pode ser explicada pela posição dos sensores no plano de simulação, pois na maioria dos trabalhos consultados durante a pesquisa, a superioridade do filtro UKF em relação ao EKF é evidenciada.

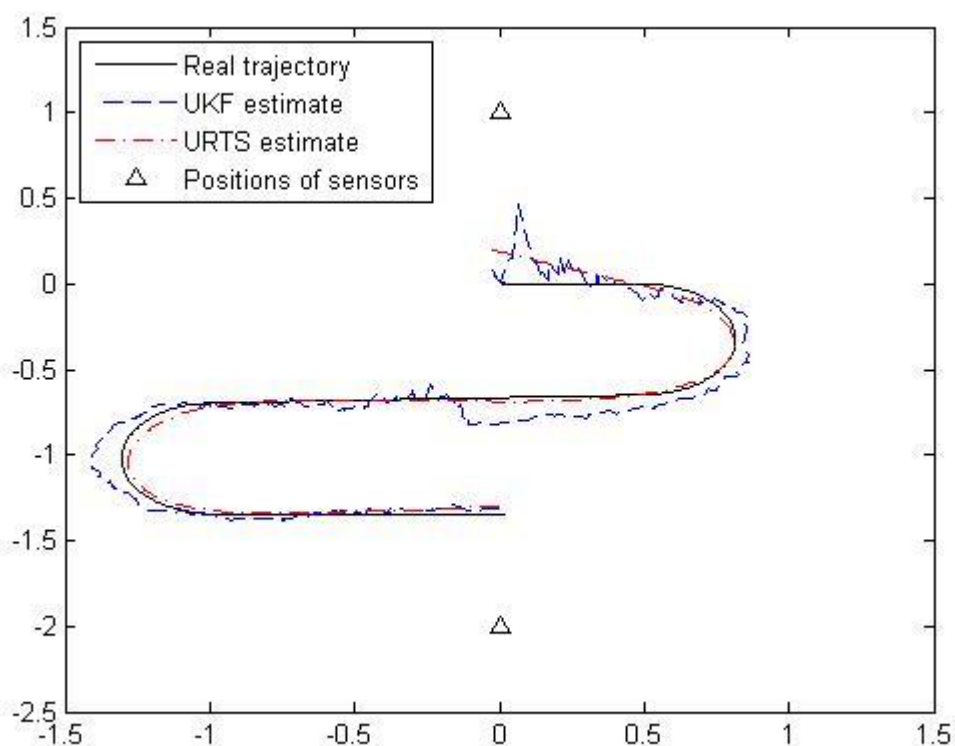


Figura 6-11 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.1).

Deve-se observar melhor essas falhas durante a predição no próximo trajeto a ser estudado, pois pode evidenciar uma pré-disposição ao melhor desempenho do EKF nos problemas de rastreamento utilizando sensores com informações angulares (*Bearing Only*).

Na Tabela 6-4 estão dispostos os erros associados a cada variante do filtro durante a execução do algoritmo no trajeto, percurso ou deslocamento do objeto proposto, no plano de trabalho simulado.

**Tabela 6-4 - Erros Posicionamento (Trajetória 1 e Pos.1)**

<b>Filtros</b>	<b>Erro (RMSE)</b>
EKF	0,0766
ERTS	0,0485
<b>EKF 2</b>	<b>0,0739</b>
<b>ERTS 2</b>	<b>0,0393</b>
UKF	0,0741
URTS	0,0445

O Filtro EKF de segunda ordem e sua variante suavizada obtiveram o melhor resultado na simulação, porém deve ser melhor analisado outros posicionamentos dos sensores no plano de trabalho simulado, para evitar possíveis interpretações errôneas devido a características do modelo.

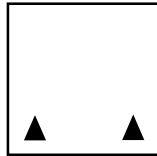
- **Trajétoria 1 Posição 2**

Foi realizada uma nova simulação alterando a posição dos sensores no plano de trabalho simulado, sendo que estas alterações podem provar uma forte dependência do modelo a posição espacial dos sensores, podendo estes influenciar em uma melhor ou pior predição do deslocamento do objeto no plano cartesiano.

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

S1=(-1;-2)

S2=(1;-2)



Durante a execução nota-se na Figura 6-12 ocorre o mesmo problema no filtro EKF durante as curvas propostas no trajeto, mesmo sendo utilizando o suavizador, o resultado se torna bem próximo ao EKF executado com os sensores na posição 1, inicialmente descartando a influência da posição dos mesmos.

A influência da posição dos sensores nas Figuras 6-12 e Figura 6-13, horizontalmente ou paralelamente ao trajeto proposto até o presente momento, demonstram pouca variação sobre resultados dos algoritmos EKF de primeira e EKF de segunda ordem e seus respectivos suavizadores.

Nota-se também grande estabilidade do movimento previsto sobre o trajeto proposto, mesmo nas previsões iniciais do modelo indicando forte propensão do modelo ao sucesso na predição desse tipo de problema de rastreamento.

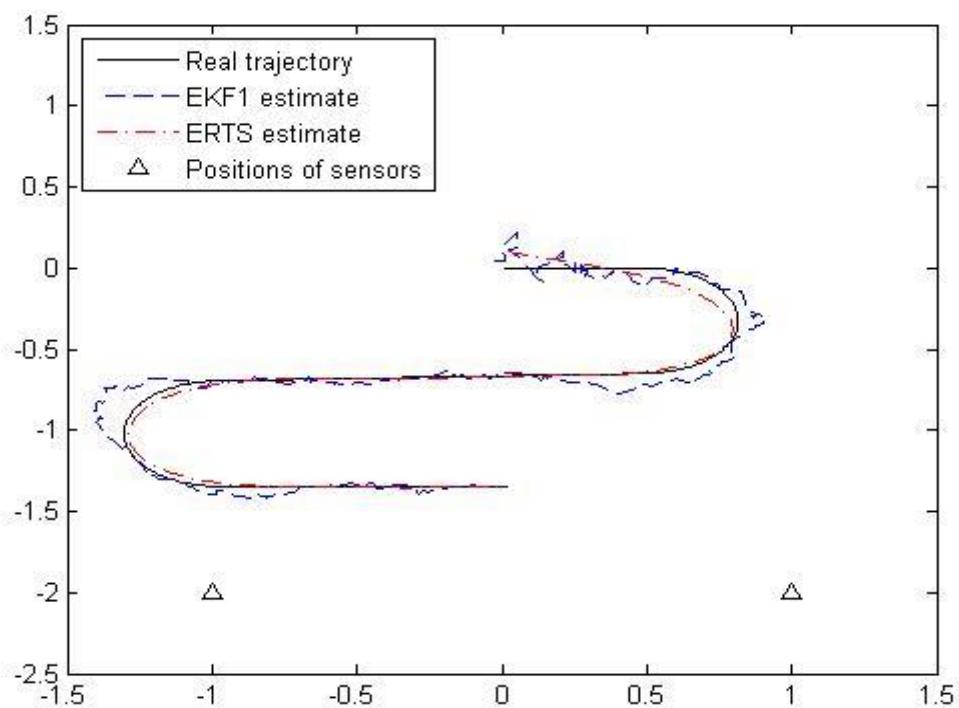


Figura 6-12 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.2).

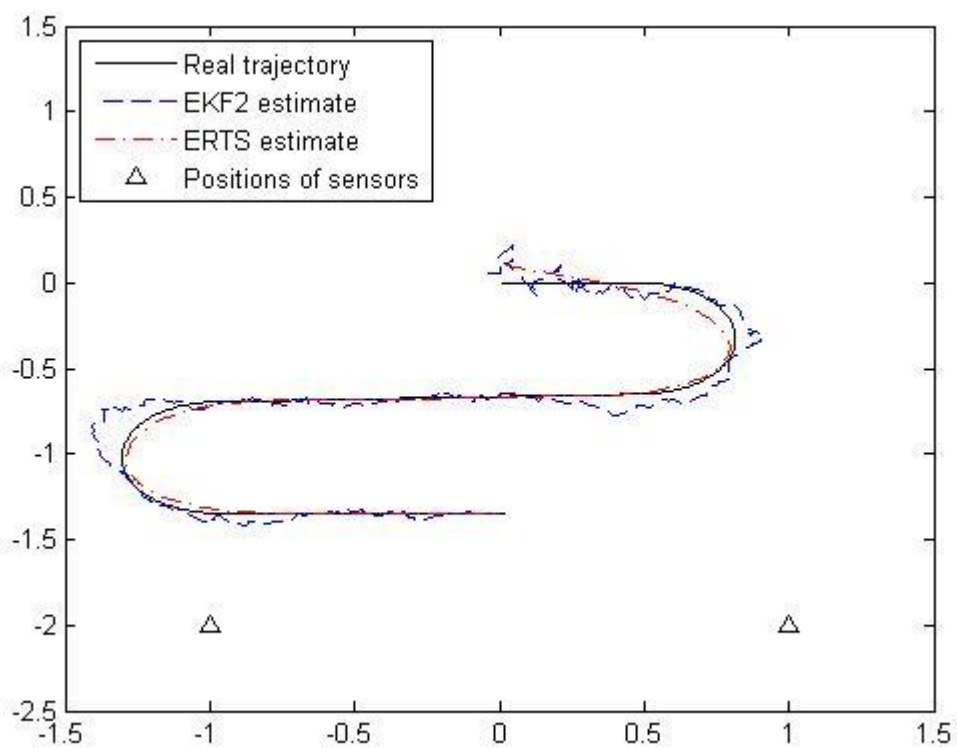
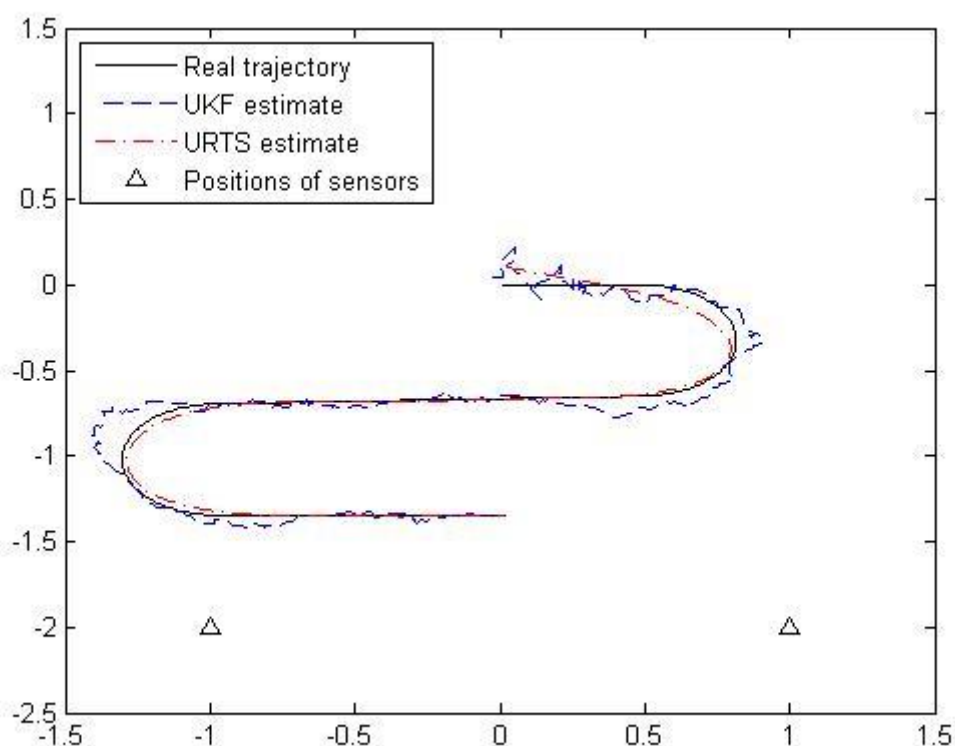


Figura 6-13 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.2).

O Filtro UKF na Figura 6-14 e seu suavizador durante a execução nessa determinada posição dos sensores sobre o plano, evidência as propriedades de adaptabilidade do filtro e forte generalização do mesmo sobre o algoritmo e o trajeto proposto.

Porém nota-se uma maior propensão ao erro do Filtro UKF durante as curvas, fenômeno esse que pode ser explicado pelo posicionamento espacial dos sensores.



**Figura 6-14 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.2).**

Na tabela 6-5 pode ser visto a proximidade dos resultados entre os Filtros EKF de primeira ordem e o UKF sendo fator determinante da escolha do melhor algoritmo para a situação proposta, os suavizadores de cada um dos Filtros e seus resultados, melhor resultado esse obtido pelo filtro UKF na sua forma suavizada.

**Tabela 6-5 - Erros Posicionamento (Trajetória 1 e Pos.2)**

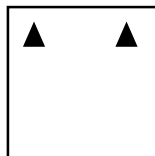
Filtros	Erro (RMSE)
EKF	0,0730
ERTS	0,0333
EKF 2	0,0735
ERTS 2	0,0334
UKF	<b>0,0730</b>
URTS	<b>0,0331</b>

### • Trajetória 1 Posição 3

Os sensores foram dispostos no sentido oposto ao realizado anteriormente comparando se existem diferenças durante as previsões iniciais e finais do trajeto ou percurso proposto.

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

$S1=(-1;1)$



$S2=(1;1)$

Durante a execução do Filtro EKF de primeira ordem descrito na Figura 5-15, a influência dos sensores e do deslocamento do objeto, se mostram de forma mais clara havendo diferenças entre a execução com os sensores na Posição 2 e Posição 3. Na Figura 5-16 pode ser visualizado o baixo desempenho do filtro EKF de segunda ordem, com maior erro durante as previsões realizadas nas curvas da trajetória e o justo do deslocamento previsto sobre o deslocamento do objeto.

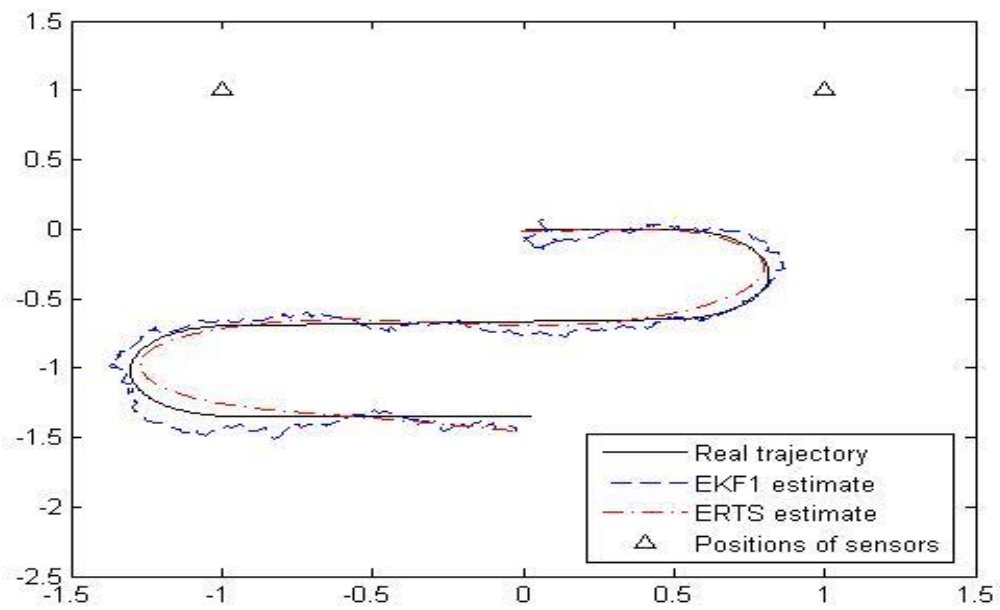


Figura 6-15 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.3).

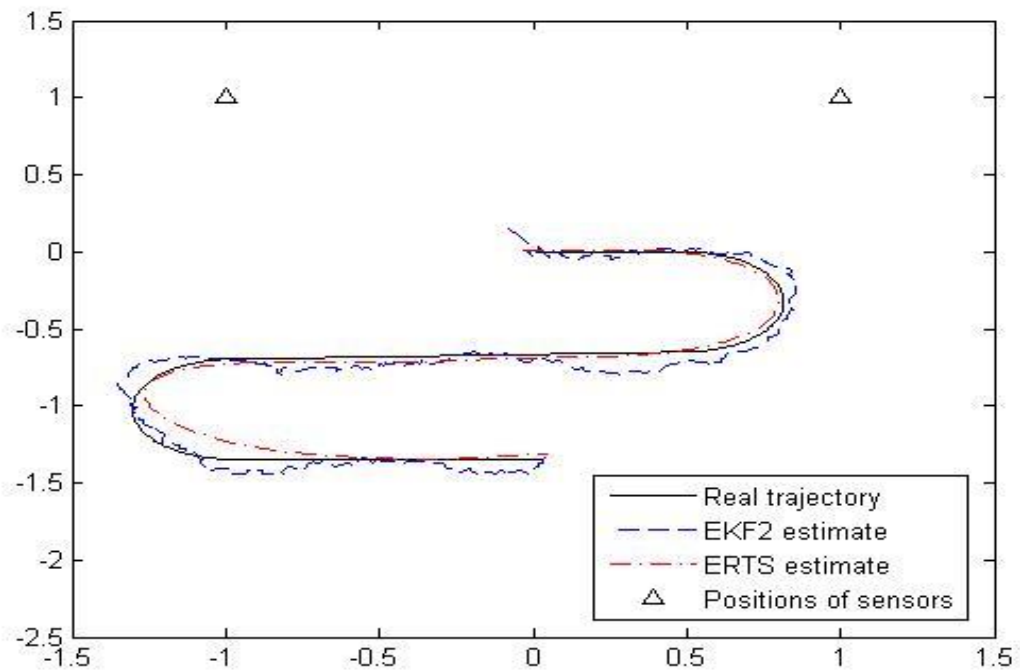
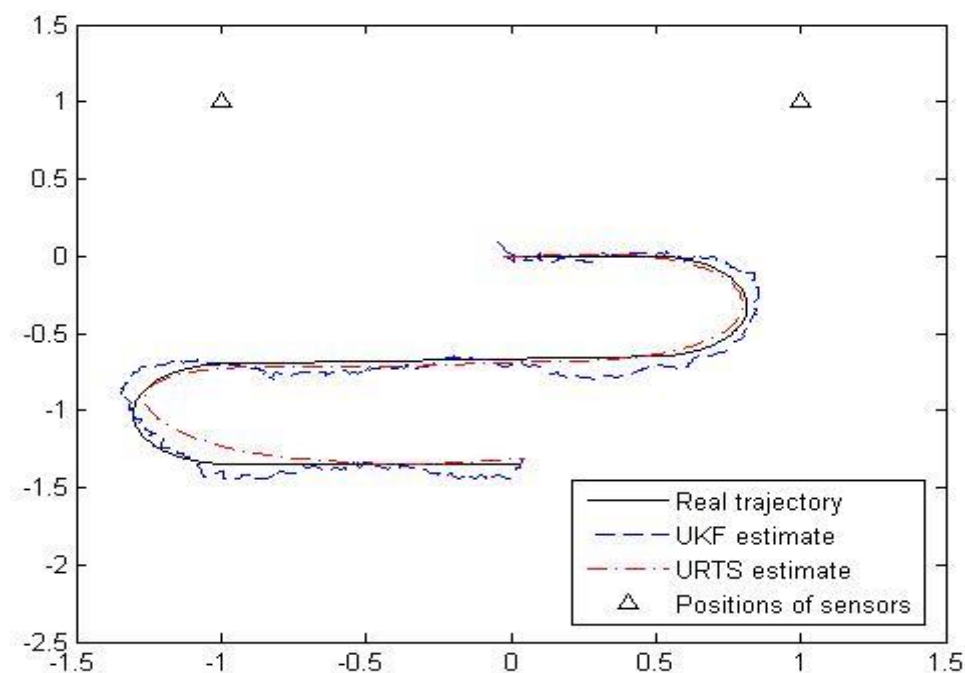


Figura 6-16 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.3).



Na Figura 5-17 durante a execução do Filtro UKF a influência da posição escalar dos sensores sobre a melhor ou pior propensão a previsões mais corretas e melhor ajuste do modelo sobre o deslocamento proposto sobre o plano de trabalho simulado.



**Figura 6-17 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.3).**

Ocorrem erros de predição até sobre pontos do percurso do objeto com deslocamento praticamente linear, a baixa performance do algoritmo UKF e seu respectivo suavizador, traz à tona um problema desconsiderado por diversos autores, a influência do trajeto cartesiano proposto e a posição relativa dos sensores com retorno angular.

Observando a Tabela 6-6, visualiza-se o baixo desempenho dos Filtros durante essa simulação.

Tabela 6-6 - Erros Posicionamento (Trajetória 1 e Pos.3)

Filtros	Erro (RMSE)
EKF	0,0790
ERTS	0,0508
EKF 2	0,0792
ERTS 2	0,0505
UKF	<b>0,0788</b>
URTS	<b>0,0504</b>

Provando a grande susceptibilidade do modelo a alteração das posições dos sensores, comparativamente se formos analisar a performance dessa variante em relação as duas efetuadas anteriormente.

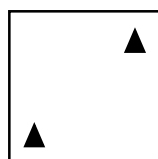
- **Trajetória 1 Posição 4**

Durante a última execução do algoritmo para o primeiro percurso proposto no trabalho, podemos concluir ao final, as dependências do modelo em relação as variáveis alteradas (posição dos sensores), e qual algoritmo de filtragem a ser utilizado para prever a posição escalar do objeto deslocado sobre o percurso simulado no ambiente de trabalho.

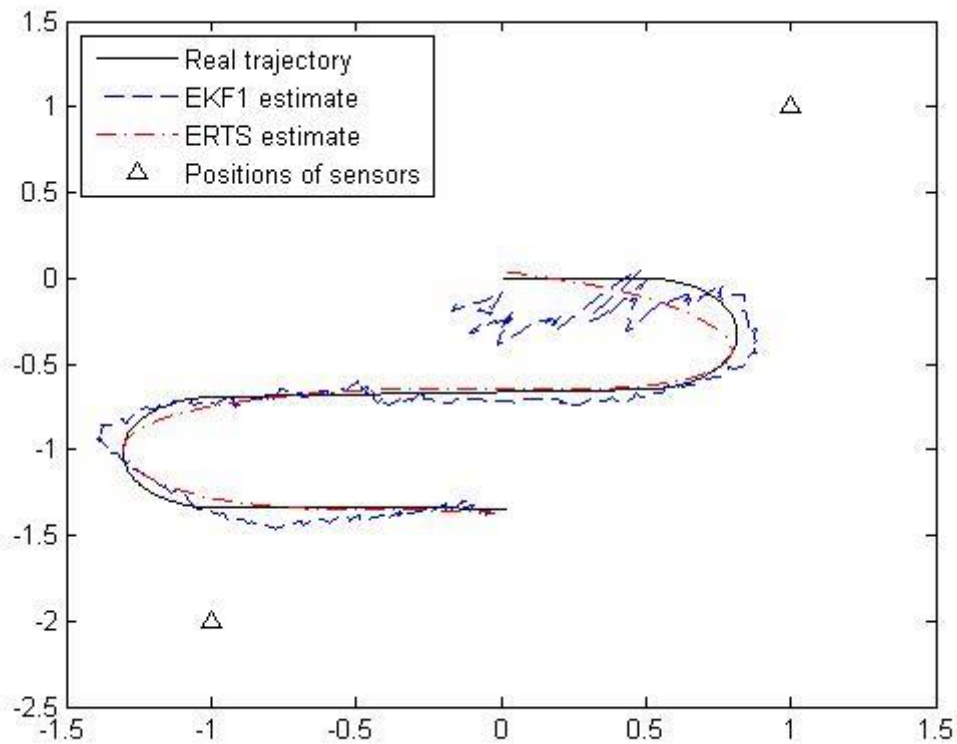
Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

S1=(-1;-2)

S2=(1;1)

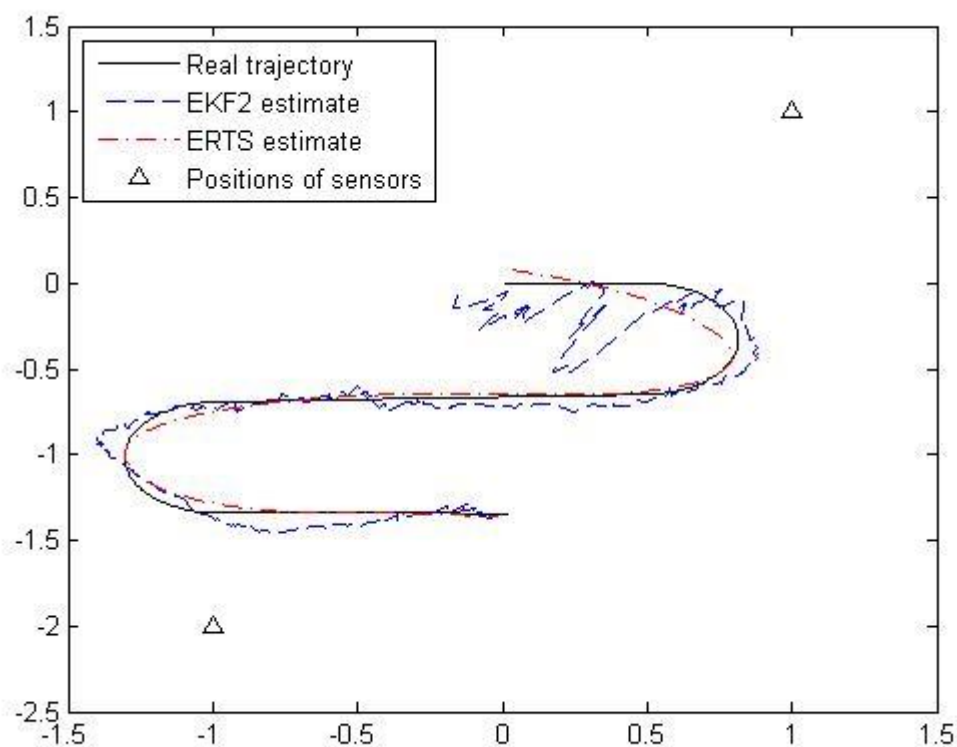


Durante a execução do filtro EKF de primeira ordem (Figura 6-18), nota-se um aumento do erro associado a predição, durante todo o trajeto proposto, grandes derivas são associadas principalmente durante o início e fim do trajeto.



**Figura 6-18 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 1 e Pos.4).**

Filtro EKF de segunda ordem (Figura 6-19), obtém uma performance inferior ao Filtro EKF de primeira ordem, fato este que não ocorreu nas três primeiras simulações, tornando clara a influência da posição dos sensores no modelo.



**Figura 6-19 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 1 e Pos.4).**

Nota-se também a baixa acurácia do modelo e relação ao trajeto ou deslocamento proposto, a posição dos sensores foi determinada de forma arbitrária, mesmo assim demonstrou uma fragilidade do modelo de rastreamento utilizando sensores de referência angular (Bearing Only).

O filtro UKF sendo citado nos trabalhos de Chaumette e Hutchinson (2006), Chen (2012), Crouse (2013), Janabi-Sharifi e Marey (2010), Haykin (2001) como uma solução viável em diversos sistemas de predição com performance superior ao EKF de primeira e segunda ordem, conforme a Figura 6-20, porém com baixo desempenho durante o percurso inicial do trajeto proposto e durante a execução do algoritmo sobre o primeiro trajeto proposto.

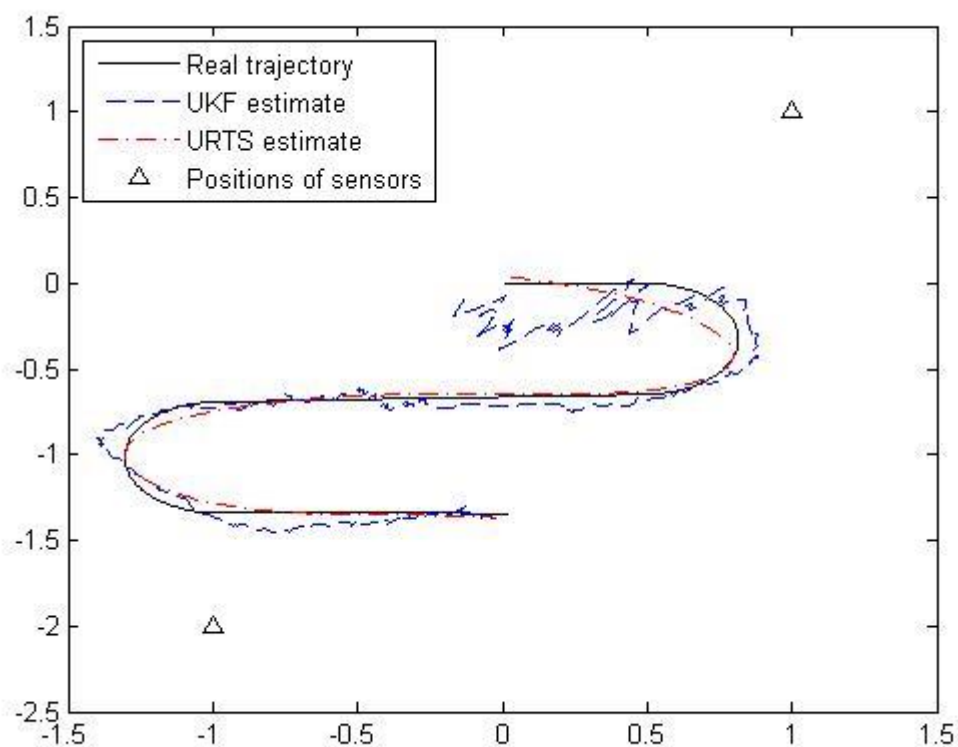


Figura 6-20 - Posicionamento utilizando o UKF e URTS (Trajetória 1 e Pos.4).

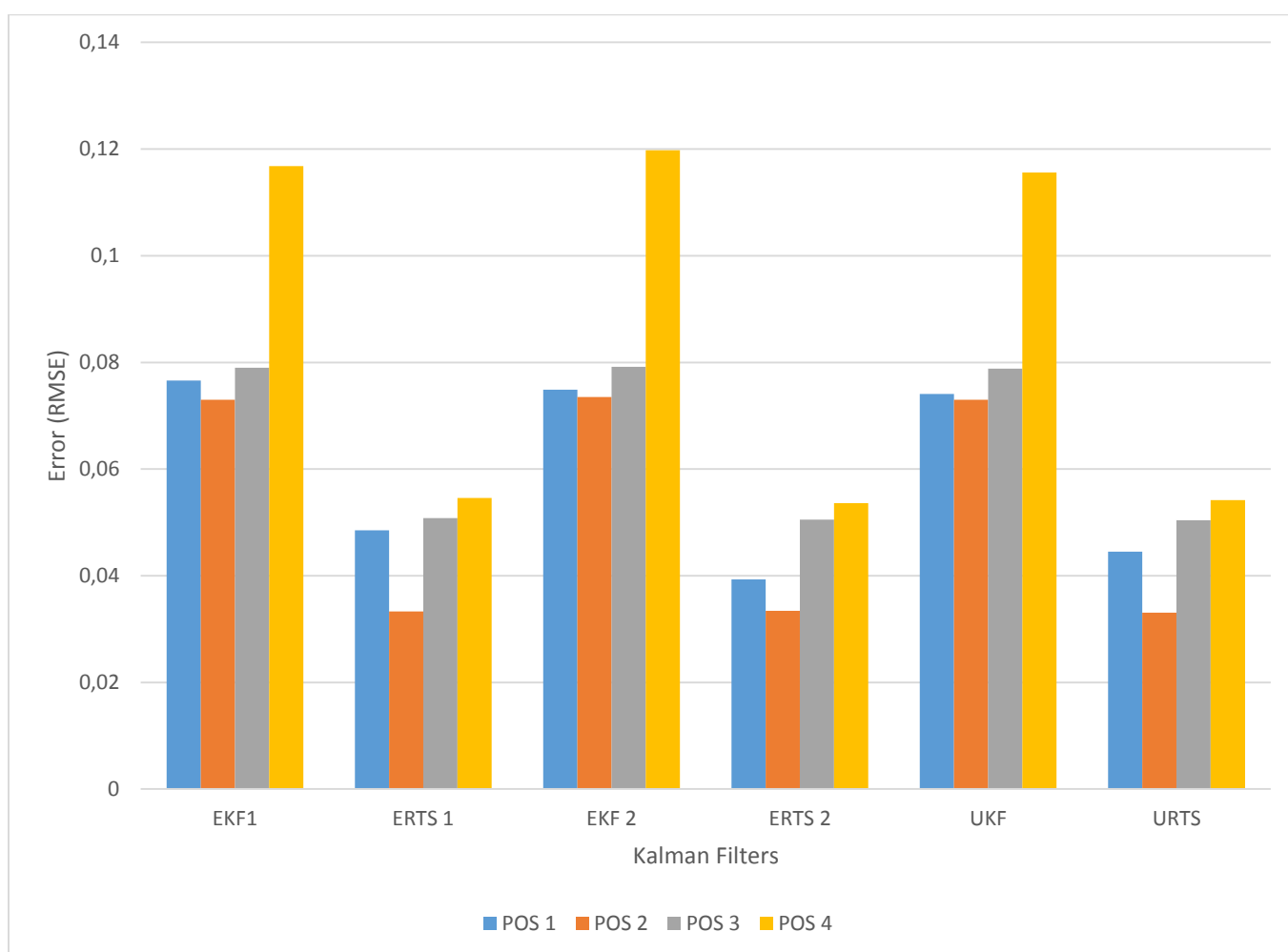
Na tabela 6-7, o maior erro associado a execução de todos os Filtros propostos, vem à tona demonstrando essa peculiaridade e fragilidade do modelo quanto a posição espacial dos sensores no modelo.

Tabela 6-7 -Erros Posicionamento (Trajetória 1 e Pos.4)

Filtros	Erro (RMSE)
EKF	0,1168
ERTS	0,0546
EKF 2	0,1198
ERTS 2	0,0536
UKF	<b>0,1156</b>
URTS	<b>0,0542</b>

Os Trajetos 2 e 3 propostos devem ser analisados, a fim de verificar essa predominância em boa parte das execuções do modelo, ou apenas ocorre sobre o Trajeto 1 simulado sobre o plano de trabalho.

Na Figura 6-21 serão expostos os valores dos erros associados a execução de cada Filtros em função da posição dos sensores com retorno angular.



**Figura 6-21 - Comparação entre a Posição dos Sensores contra Erros (Trajetória 1).**

Nota-se que durante a execução da filtragem Tabelas 6-4, 6-5, 6-6, 6-7, afim de obter posição e velocidade do objeto deslocado sobre o plano de trabalho simulado, o melhor desempenho nas 4 posições definidas para os sensores de rolamento, é o Filtro UKF com suavização seguido pelo Filtro EKF de segunda ordem com suavização.

Tendo o pior desempenho no primeiro trajeto vem o Filtro EKF de primeira ordem e sua variante suavizada.

Tomando como variável posição dos sensores de rolamento, pode-se concluir que na Posição 4, os Filtros oferecem desempenho inferior e perda de acurácia durante sua execução, denotando uma tendência a erros associados a posição dos sensores.

### 6.3.2 Resultados (Deslocamento Circular)

- **Trajectoria 2 Posição 1**

Na Figura 6-22 é demonstrada a trajetória real utilizada na estimativa do método, estando representada a trajetória e posicionamento dos sensores, referente as simulações que se seguem.

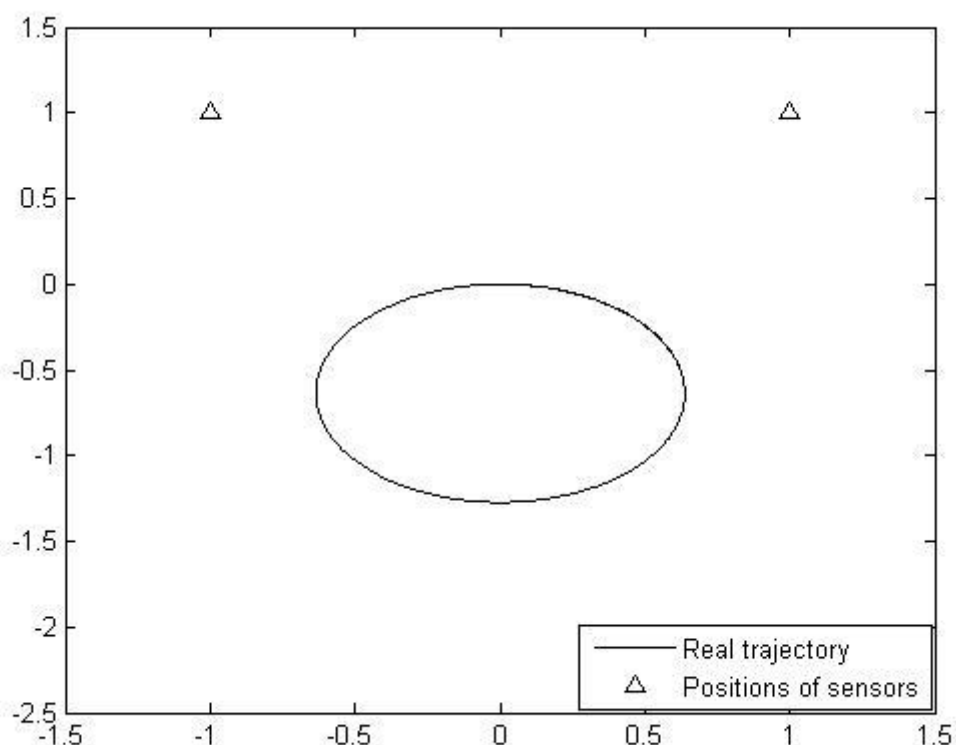
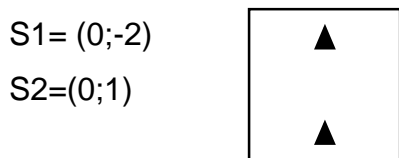
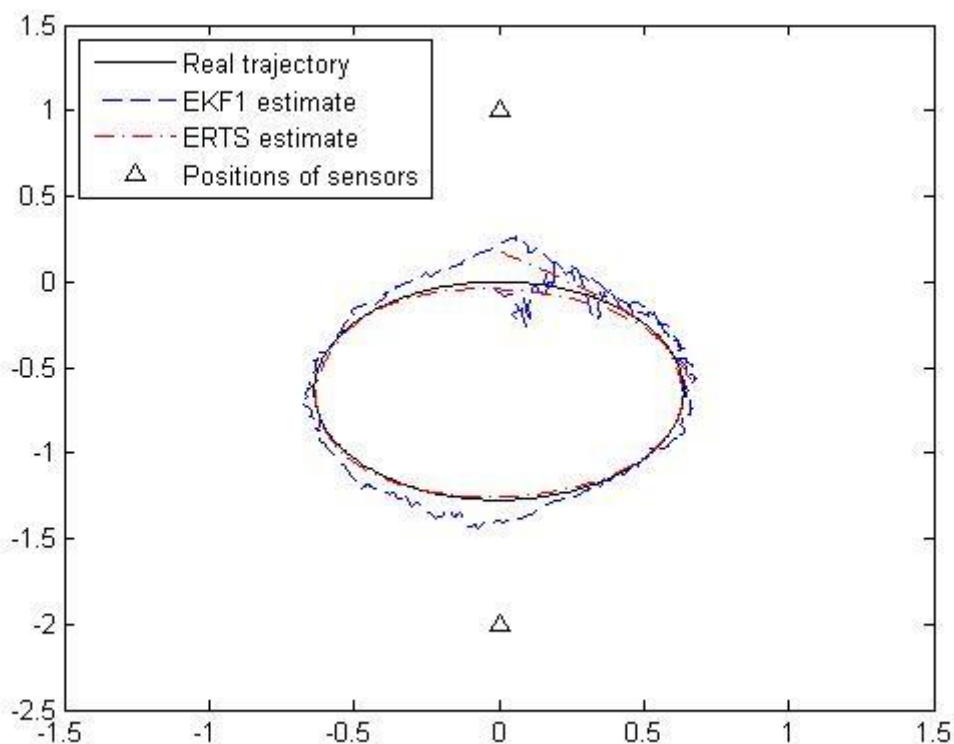


Figura 6-22 - Trajeto Circular

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:



Durante a execução do Filtro EKF de primeira ordem em um trajeto circular, nota-se claramente na Figura 6-23, a baixa acuracidade da predição e pouca precisão na determinação da posição e velocidade do objeto no trajeto simulado.

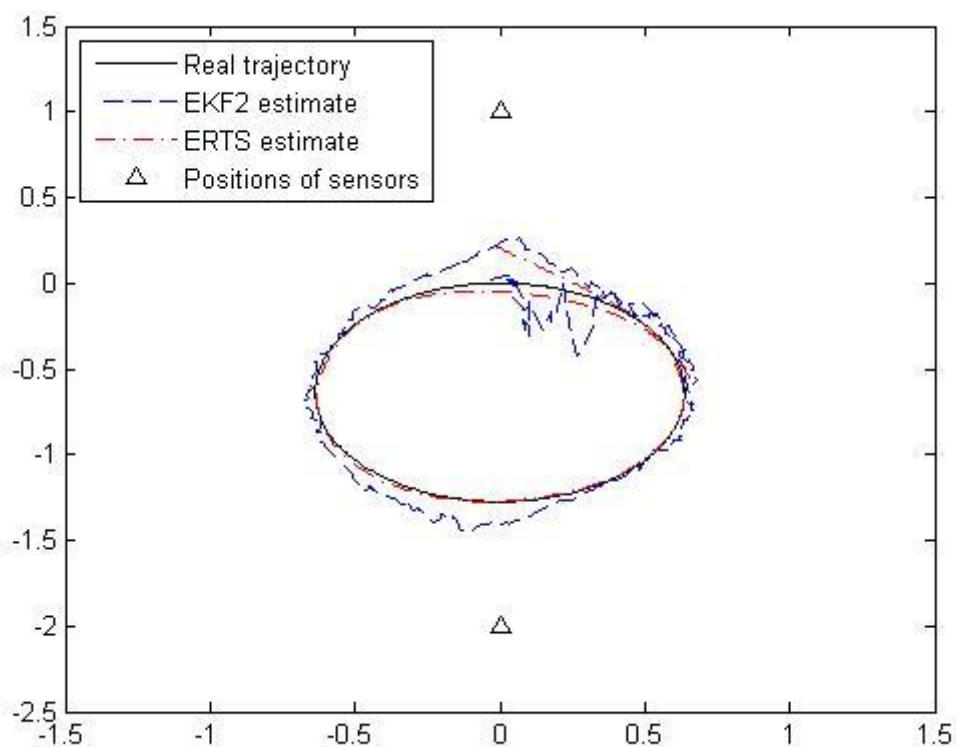


**Figura 6-23 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.1).**

Algo que não foi tão perceptível nas simulações anteriores sobre o trajeto linear com curvas, fato que pode estar ocorrendo pela dificuldade de predizer a velocidade do objeto sobre o percurso e/ou trajeto simulado.

Pode ser visualizado na Figura 6-24 a dificuldade da predição do Filtro EKF de segunda ordem durante sua execução, durante o início do percurso pode se notar erros associados na determinação da trajetória do objeto.



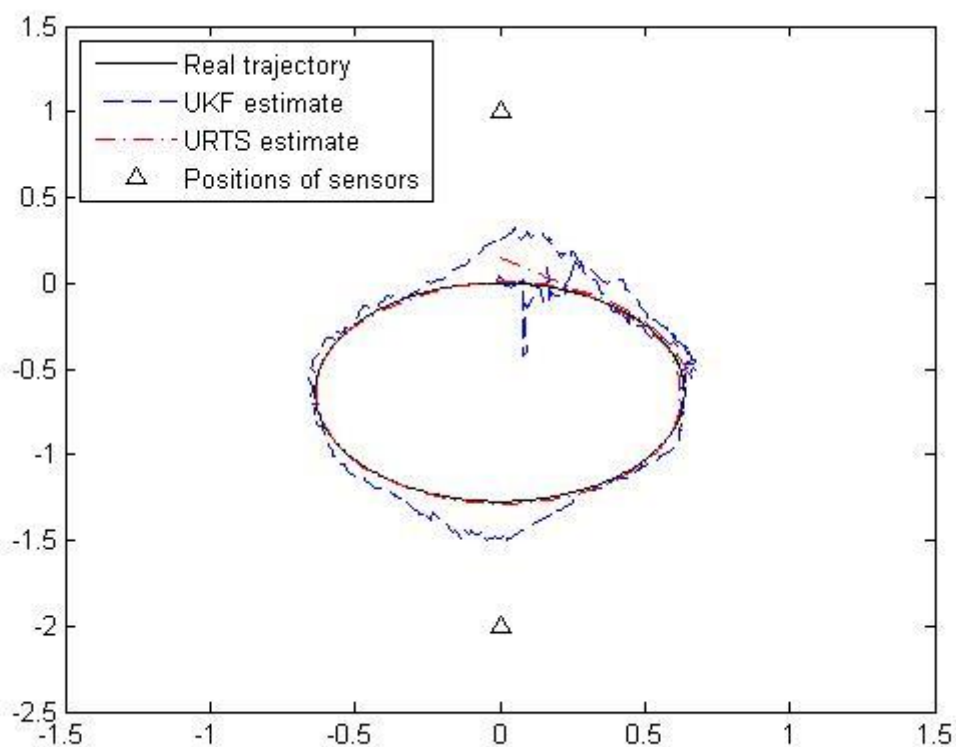


**Figura 6-24 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.1).**

Porém a variante suavizada do Filtro EKF conseguiu obter uma boa generalização do percurso realizado pelo objeto, obtendo acurácia na determinação da velocidade e posição.

Os resultados das primeiras simulações trazem à tona uma peculiaridade do modelo utilizando sensores angulares, o percurso tem forte influência sobre a previsão realizada pelos filtros.

A execução do Filtro UKF conforme mostrado na Figura 6-25, expõe a dificuldade do filtro em generalizar e determinar a posição e velocidade do objeto, com erros associados a determinação do percurso.



**Figura 6-25 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.1).**

Como a trajetória não tem perfil completamente circular, devido a sua construção isso pode estar influenciando sobre a previsão.

A interação dos parâmetros posição dos sensores e trajetória deve ser melhor analisada durante as outras execuções, tentando evidenciar uma relação entre esses dois parâmetros e a previsão do trajeto proposto.

Na tabela 6-8 pode ser verificado o melhor desempenho do Filtro UKF e sua variante suavizada, porém as Figuras 5-23, 5-24 e 5-25 expõem, a baixa adaptabilidade dos Filtros EKF de primeira e segunda ordem e UKF na determinação do trajeto proposto em função da velocidade e posição do objeto.

Tabela 6-8 - Erros Posicionamento (Trajetória 2 e Pos.1)

Filtros	Erro (RMSE)
EKF	0,0951
ERTS	0,0328
EKF 2	0,1040
ERTS 2	0,0405
UKF	<b>0,0948</b>
URTS	<b>0,0316</b>

Sendo necessário a utilização do Filtros em suas formas suavizadas, a fim de obter generalização do trajeto e bom ajuste dos dados filtrados em comparação a trajetória simulada.

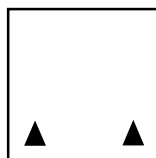
- **Trajétória 2 Posição 2**

Foi realizada uma nova simulação alterando a posição dos sensores no plano de trabalho simulado, sendo que estas alterações podem comprovar uma forte dependência do modelo a posição espacial dos sensores, podendo estes influenciar em uma melhor ou pior predição do deslocamento do objeto no plano cartesiano.

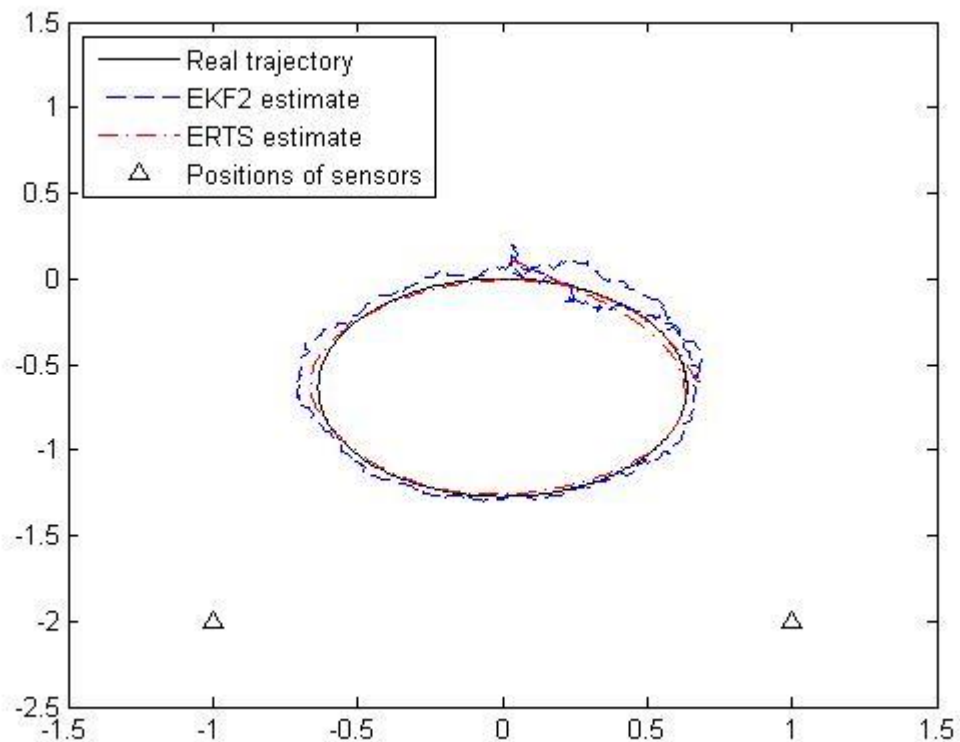
Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

S1=(-1;-2)

S2=(1;-2)



Na filtragem realizada pelo EKF de primeira ordem, nota-se erros ocorrendo na aproximação do modelo quanto a trajetória (Figuras 6-26), podendo ser explicado pelo fato do modelo não conseguir uma boa aproximação do modelo em relação a velocidade do objeto quando em deslocamento circulares.



**Figura 6-26 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.2).**

Durante a realização da simulação utilizando o Filtro EKF de segunda ordem, os mesmos padrões de erros associados a predição do modelo quanto ao deslocamento do objeto ocorrem como pode ser visto nas Figuras 6-26 e 6-27 mesmo com os sensores localizados a 90°.

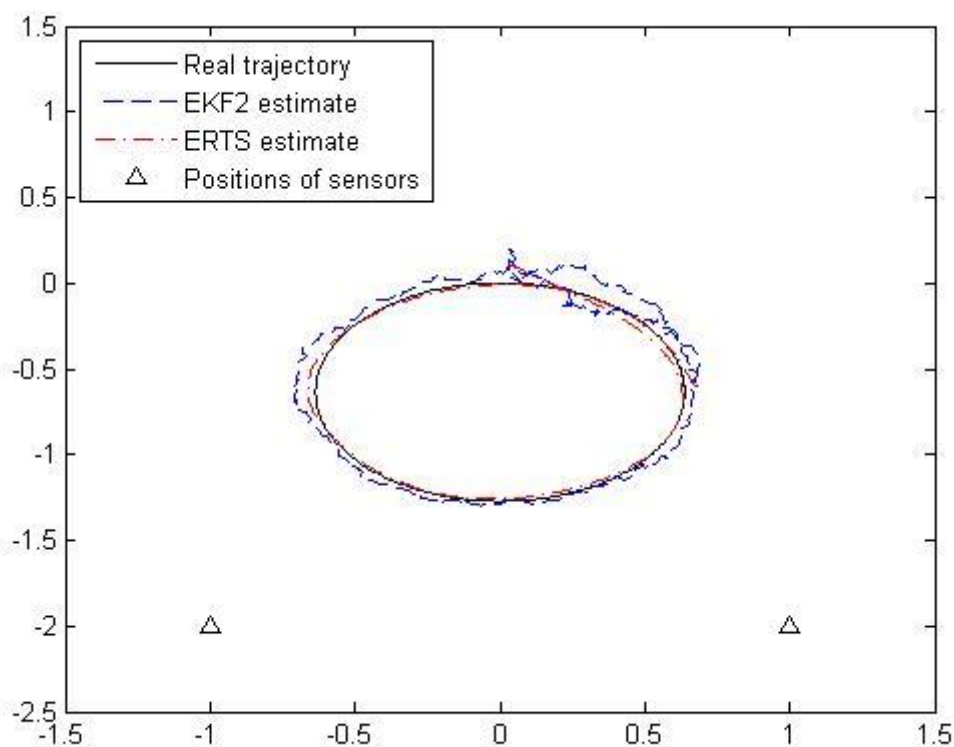


Figura 6-27 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.2).

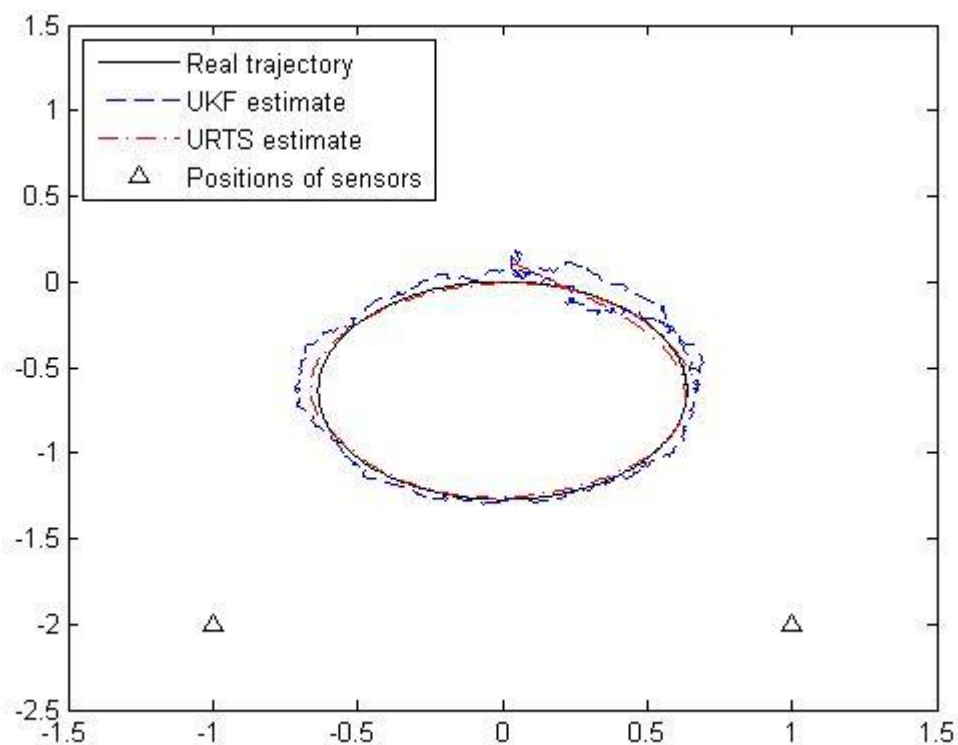


Figura 6-28 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.2).

Porém a variante dos Filtros EKF de primeira e segunda ordem suavizados, obtém boa generalização da velocidade e posição do objeto como pode ser observado nas Figuras 6-26 e 6-27, fato esse explicado pela facilidade do RTS (H. E. RAUCH, STRIEBEL e TUNG, 1965) com sua característica bayesiana estatística, em se utiliza de dois Filtros (Cálculo inverso do modelo dinâmico posterior e cálculo anterior a filtragem utilizando o modelo dinâmico).

Na tabela 6-9 é demonstrado o melhor desempenho do Filtro EKF de primeira, tendo resultado aproximados as variantes EKF de segunda ordem e UKF.

Fato esse que comprova a boa generalização dos algoritmos na tarefa de determinação de posição e velocidade.

**Tabela 6-9 - Erros Posicionamento (Trajetória 2 e Pos.2)**

<b>Filtros</b>	<b>Erro (RMSE)</b>
<b>EKF</b>	<b>0,0639</b>
<b>ERTS</b>	<b>0,0263</b>
<b>EKF 2</b>	0,0641
<b>ERTS 2</b>	0,0265
<b>UKF</b>	0,0641
<b>URTS</b>	0,0263

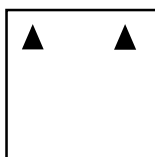
- **Trajatória 2 Posição 3**

Os sensores foram dispostos no sentido oposto ao realizado anteriormente comparando se há diferenças durante as previsões iniciais e finais do trajeto ou percurso proposto.

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

$$S1=(-1;1)$$

$$S2=(1;1)$$



Na execução do EKF de primeira ordem com os sensores posicionados paralelamente no sentido oposto da posição 2, pode-se visualizar claramente na Figura 6-29 um maior erro associado a aproximação durante a realização da previsão do trajeto no meio do percurso.

Sendo as trajetórias previstas nas Figuras 6-29, 6-30, 6-31 compostas por sistemas com acelerações, ela não é completamente circular sendo composta por arcos, essas composições da trajetória a ser prevista, aliado a posição dos sensores com retorno angular formam um componente crítico na predição do trajeto proposto.

Mesmo o Filtro EKF de segunda ordem exibido na Figura 6-30 oferece um baixo desempenho na previsão dessa trajetória proposta, os arcos componentes da trajetória e diferentes acelerações causadas voluntariamente durante a criação do percurso causam grande problemas e trazem à tona a uma fragilidade do modelo em determinar velocidade de objetos em percursos com variação de velocidade ao longo do trajeto.

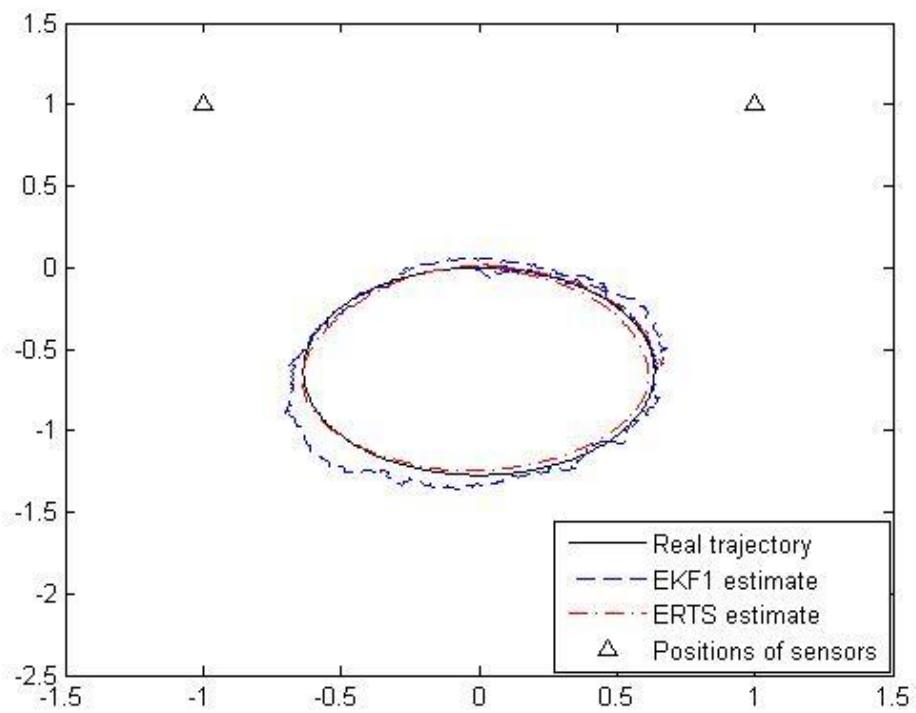


Figura 6-29 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.3).

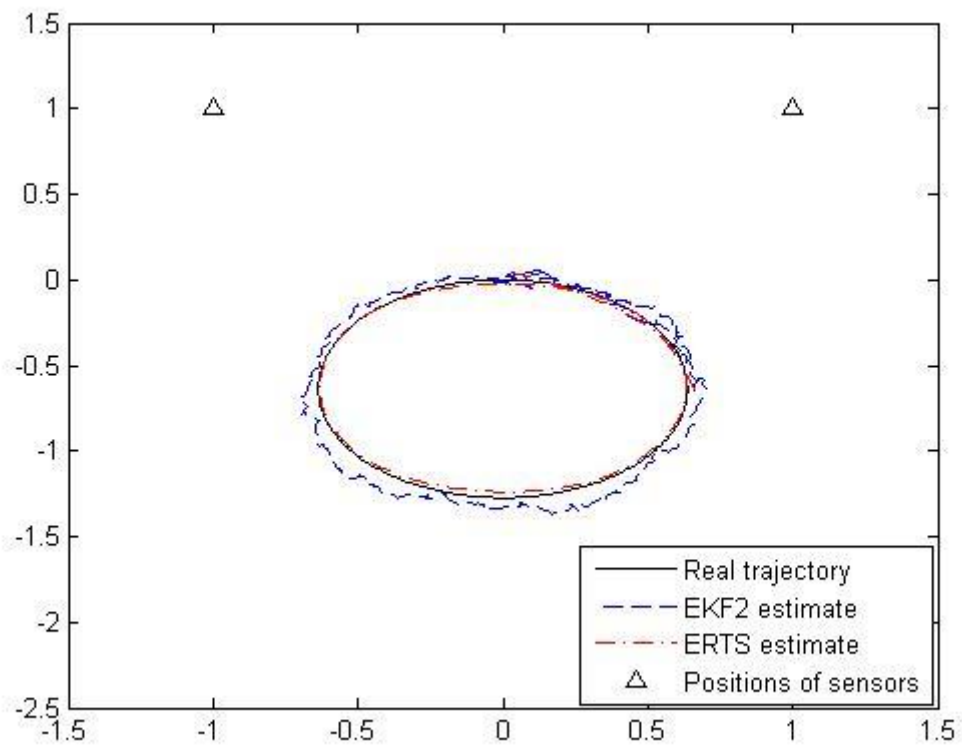




Figura 6-30 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.3).

As variantes do filtro EKF de primeira e segunda ordem, novamente obtém bons resultados na generalização e aproximação do modelo e do percurso trajeto, conseguindo obter boa performance na estimação da velocidade e posição do objeto.

Pode-se notar mesmo na execução do Filtro UKF na Figura 6-31 a dificuldade na predição do deslocamento do objeto com erros no meio do percurso proposto ao objeto sobre o plano de trabalho simulado.

Porém existe um bom ajuste do modelo utilizando o Filtro UKF com suavizador na predição da trajetória e/ou percurso realizado pelo objeto, tornando a filtragem um bom recurso a ser utilizado nas tarefas de rastreamento.

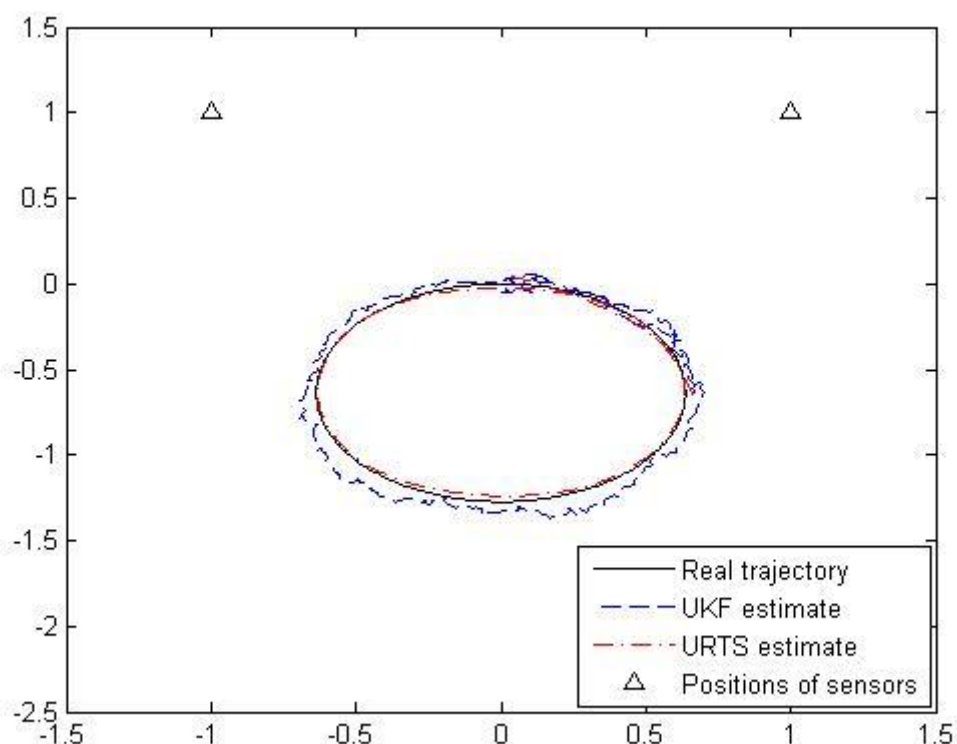


Figura 6-31 -Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.3).

Na tabela 6-10 são expressos os erros de cada um dos Filtros utilizados, o menor erro foi obtido pelo Filtro EKF de primeira ordem e sua variante suavizada, seguido bem próximo pelo Filtro UKF e sua variante suavizada, realizando novas simulação devido à proximidade dos valores, essas performances podem ser alternadas, tendo outro filtro elencado como tendo obtido melhor desempenho na estimação.

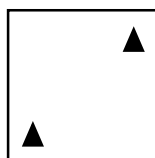
**Tabela 6-10 - Erros Posicionamento (Trajetória 2 e Pos.3)**

<b>Filtros</b>	<b>Erro (RMSE)</b>
<b>EKF</b>	<b>0,0680</b>
<b>ERTS</b>	<b>0,0267</b>
<b>EKF 2</b>	0,0681
<b>ERTS 2</b>	0,0269
<b>UKF</b>	0,0682
<b>URTS</b>	0,0267

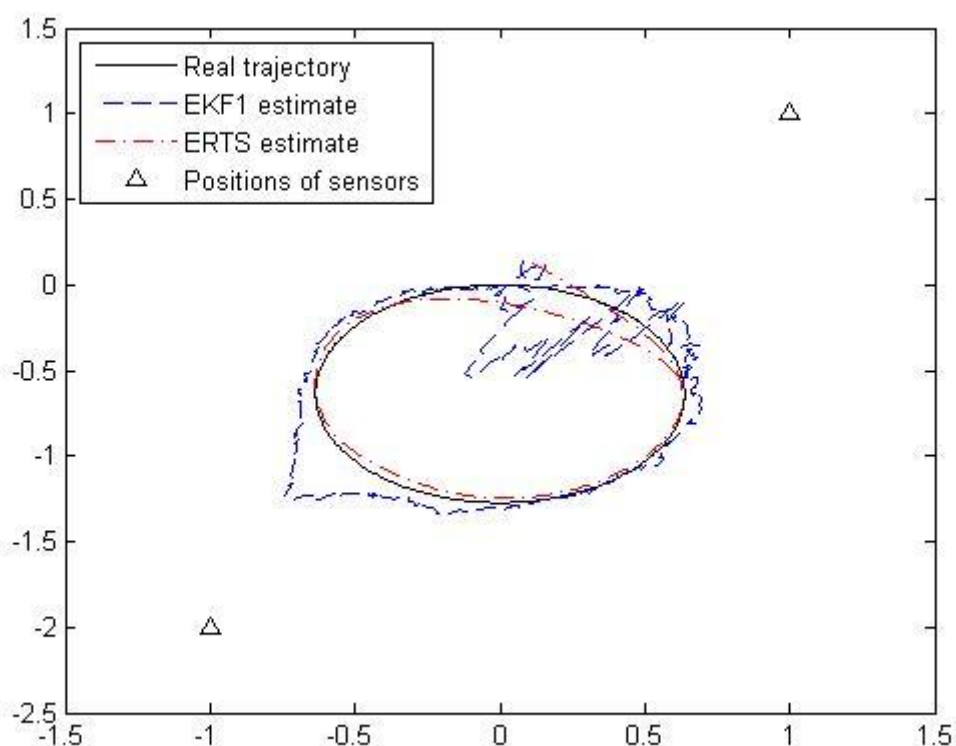
- **Trajetória 2 Posição 4**

Durante a execução do algoritmo para o segundo percurso proposto no trabalho, pode-se concluir inicialmente uma forte tendência do modelo a erros quando as dependências do modelo em relação as variáveis alteradas (posição dos sensores) e concluir qual a melhor posição e algoritmo de filtragem a ser utilizado para prever a posição escalar do objeto deslocado sobre o percurso simulado no ambiente de trabalho.

- $S1=(-1;-2)$
- $S2=(1;1)$

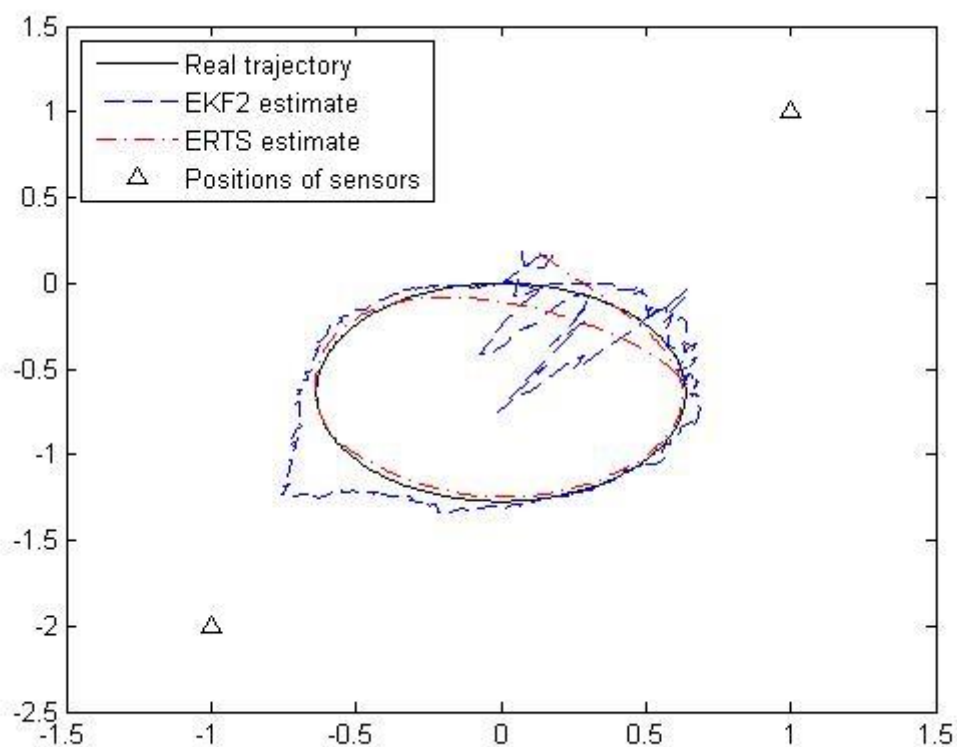


Na Figura 6-32, pôde ser verificada a execução do Filtro EKF de primeira ordem e segunda ordem e suas variantes suavizadas, ocorre nessa simulação um aumento significativo do erro na predição da trajetória do objeto, erro este que pode ser explicado pela dificuldade do algoritmo em prever a trajetória do objeto para determinar sua posição e velocidade dada a posição dos sensores de rolamento (angulares).



**Figura 6-32 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 2 e Pos.4).**

Os filtros nas Figura 6-33 e 6-34 denotam a grande influência da posição cartesiana dos sensores, influência esta pouco representativa nas simulações anteriores.



**Figura 6-33 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 2 e Pos.4).**

A dificuldade na predição da trajetória com características circulares demonstrada nas simulações efetuadas anteriormente, estavam demonstrando uma característica do modelo em ter dificuldade em predições nesse tipo de trajetória.

Porém as informações sobre os erros estavam se mantendo próximas, a partir da simulação realizada com o sensor na posição 4, foi notada grande influência da posição sensorial sobre o percurso realizado pelo objeto.

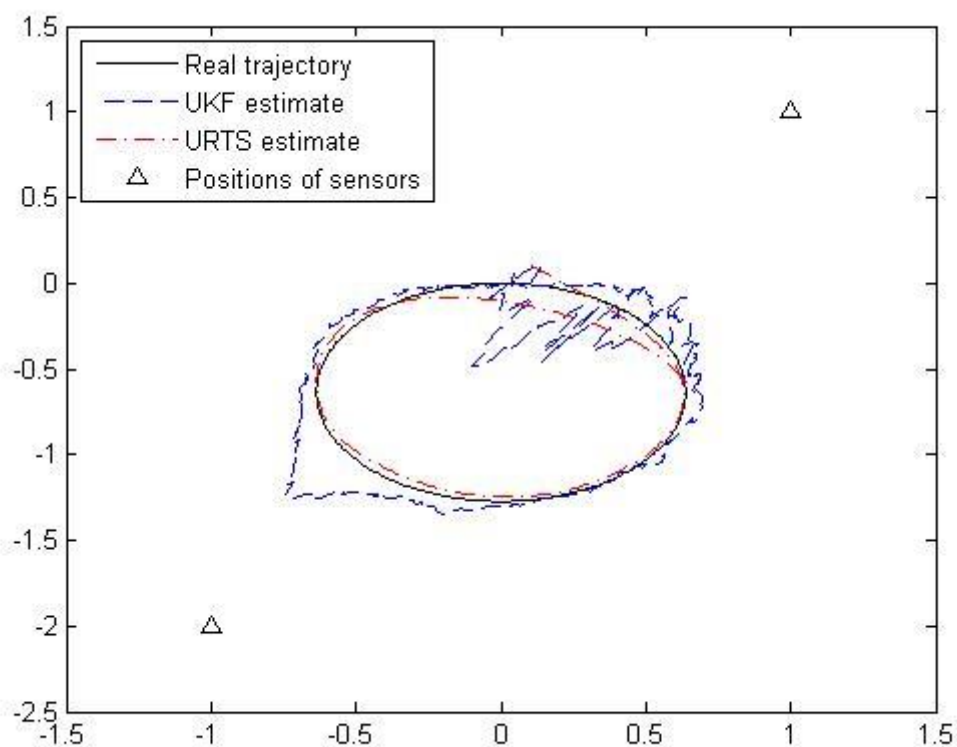


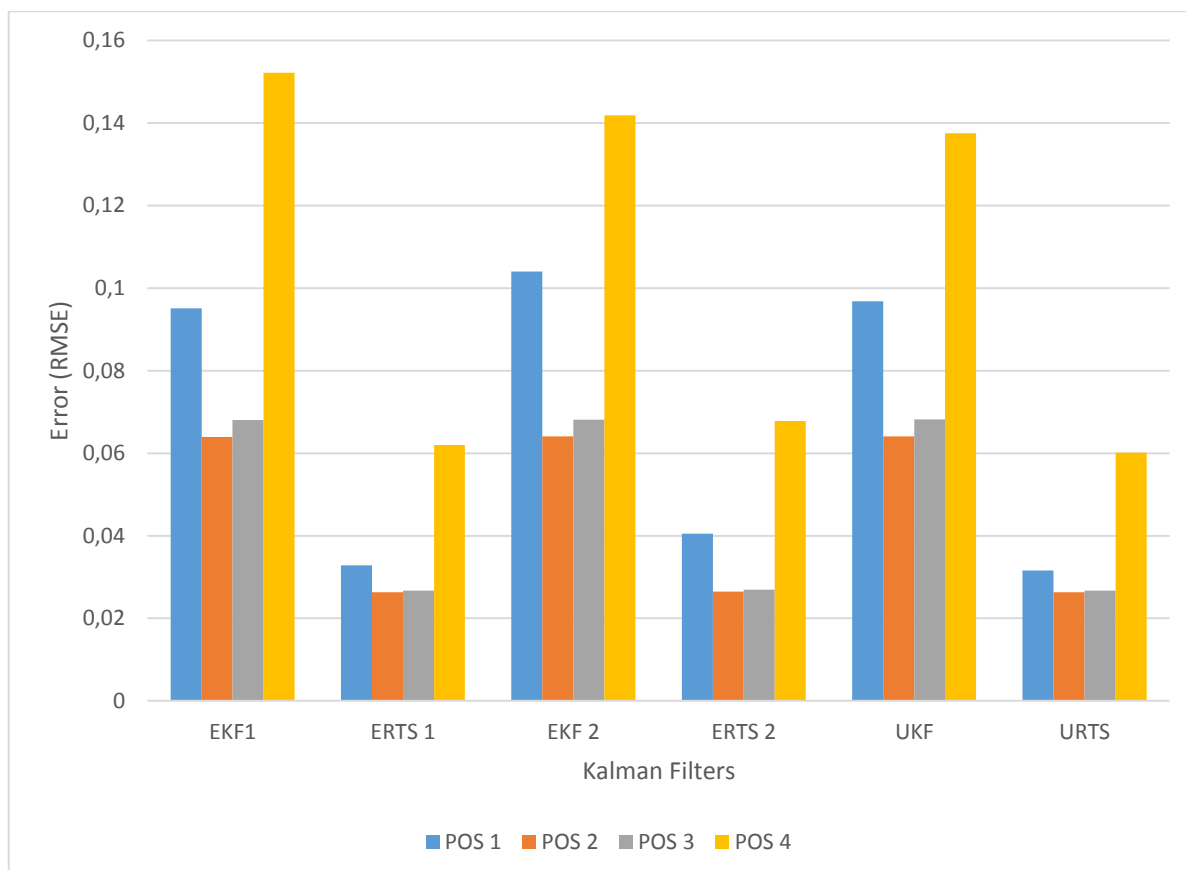
Figura 6-34 - Posicionamento utilizando o UKF e URTS (Trajetória 2 e Pos.4).

A Tabela 6-11 corrobora as informações descritas acima na influência da posição dos sensores sobre o modelo de predição, a posição 4 traz consigo um aumento significativo do erro na predição.

Tabela 6-11 - Erros Posicionamento (Trajetória 2 e Pos.4)

Filtros	Erro (RMSE)
EKF	0,1522
ERTS	0,0620
EKF 2	0,1419
ERTS 2	0,0678
<b>UKF</b>	<b>0,1375</b>
<b>URTS</b>	<b>0,0601</b>

A dificuldade na predição pode ser visualizada durante as simulações, as variações da trajetória durante a formação do trajeto simulado, torna as determinações de posição e velocidade do objeto imprecisas.



**Figura 6-35- Comparação entre a Posição dos Sensores x Erros (Trajetória 2).**

Na Figura 6-35 nota-se que segunda trajetória demonstrou uma falta de acurácia na previsão de deslocamento sobre percursos circulares pelo Filtros EKF de primeira e segunda ordem, porém com o suavizador de ambos os filtros, foi conseguido pelo método bom ajustes do modelo ao trajeto simulado.

Inicialmente foi descartada a influência da posição dos sensores de rolagem como um dos fatores determinantes quanto a eficácia da previsão do modelo, durante a simulação ficou evidenciado na Posição 4 uma forte tendência ao erro na determinação de posição e velocidade do objeto deslocado sobre o percurso proposto.

### 6.3.3 Resultados (Deslocamento com Mudanças Bruscas de Direção).

- **Trajétoria 3 Posição 1**

Na última simulação realizada durante este trabalho, o trajeto simulado é composto por mudanças bruscas de direção, com componentes de aceleração nas curvas, sendo visualizado o trajeto simulado do objeto na Figura 6-36

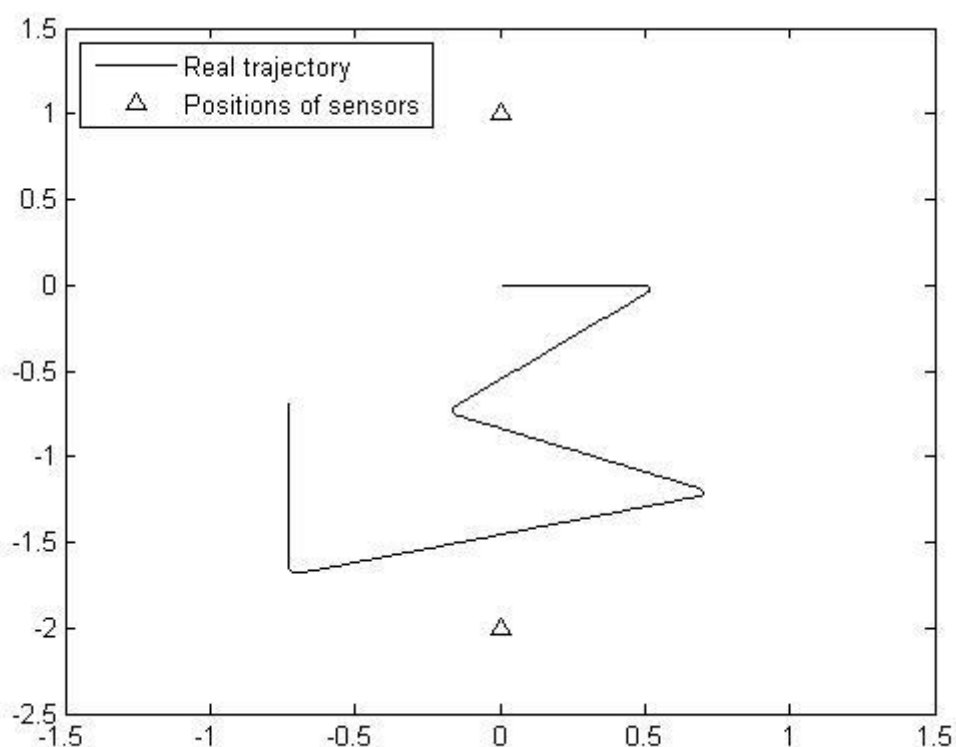


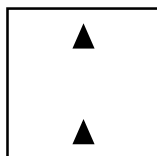
Figura 6-36 – Trajeto com mudanças bruscas de direção.

O principal objetivo dessa simulação é determinar performance do método em determinar a posição e velocidade do objeto no plano de trabalho tendo um trajeto composto de difícil percepção e diferentes posições dos sensores de rolamento (ângulos).

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

S1= (0;-2)

S2=(0;1)



Na Figura 6-37 pode ser notada a boa generalização realizada pelo Filtro EKF de 1 ordem e sua variante suavizada em determinar a posição do objeto sobre a trajetória proposta.

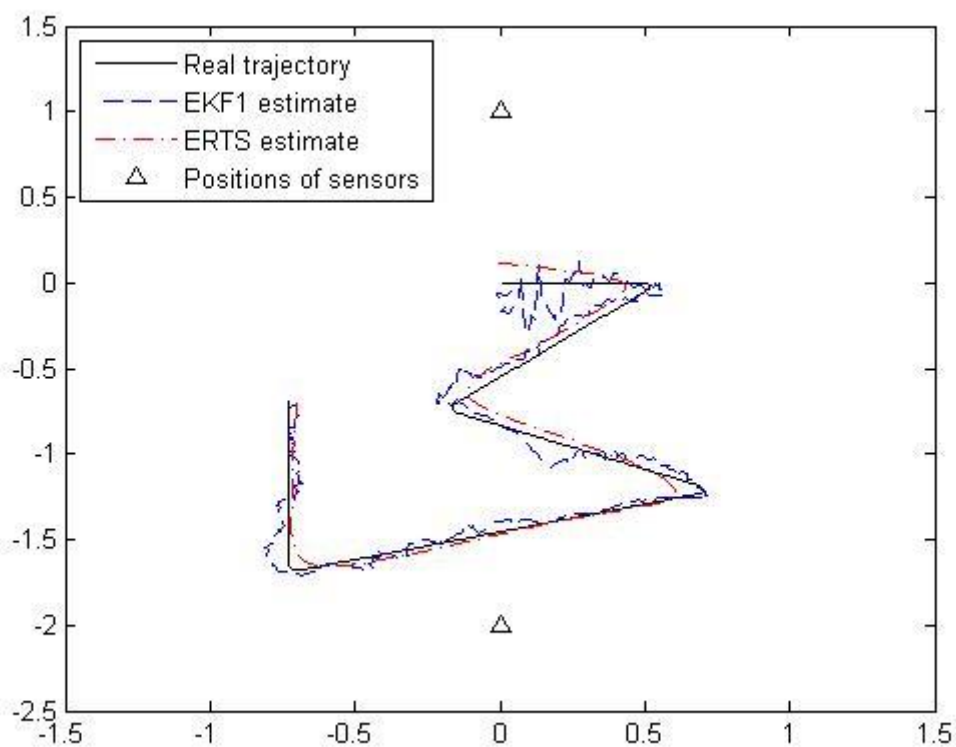
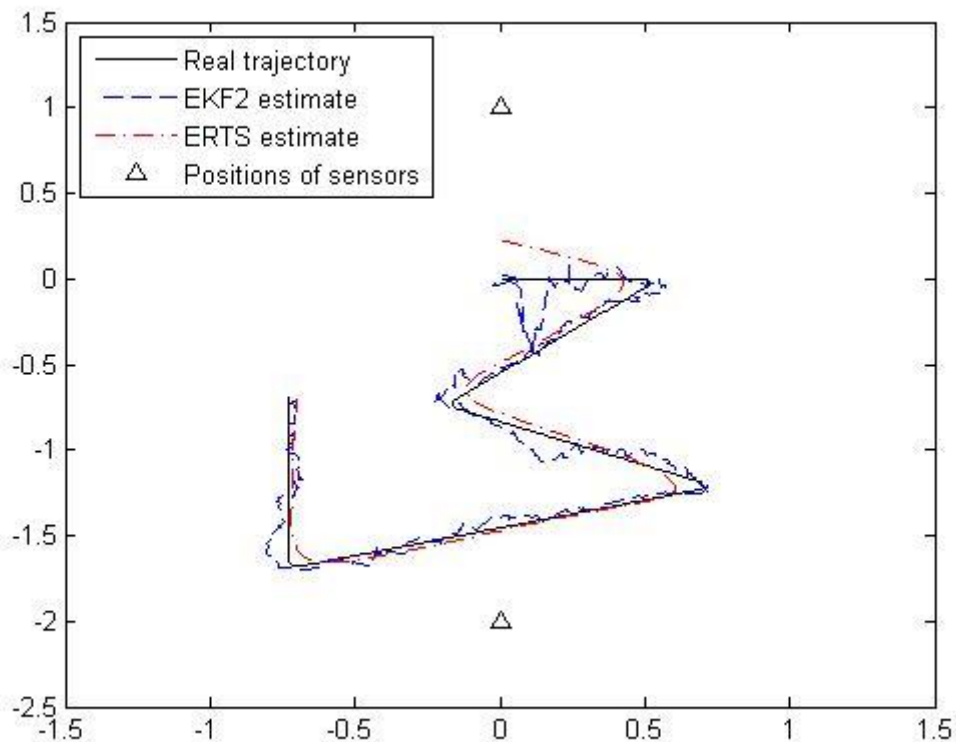


Figura 6-37 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.1).



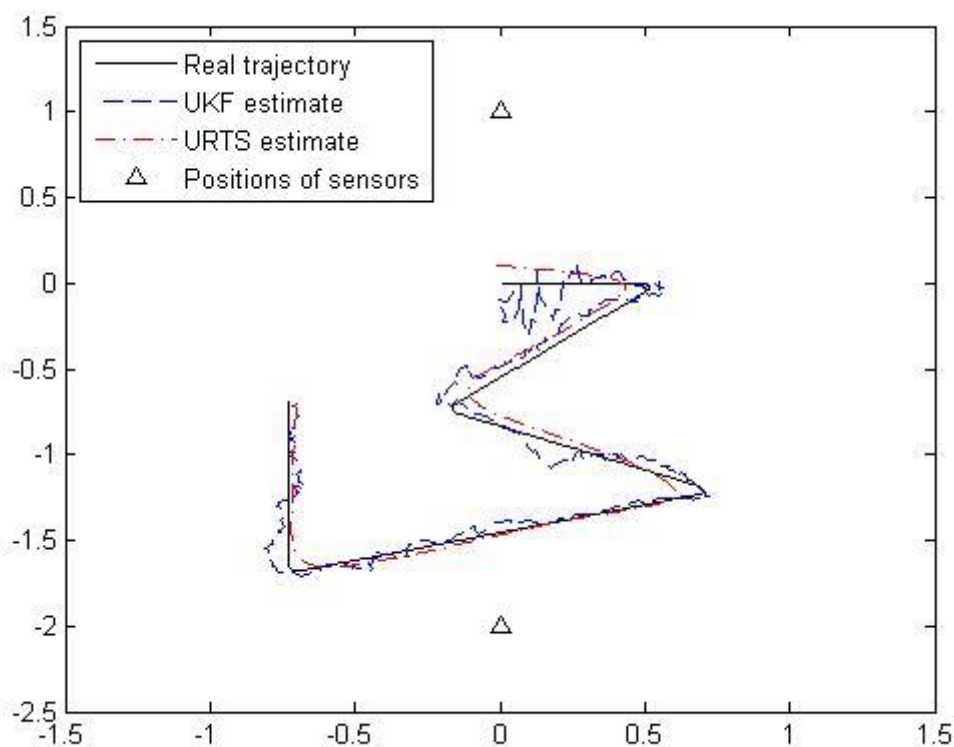
Houve um menor índice de acerto do modelo no início do trajeto, fato esse devido a adaptação inicial das covariâncias do Filtro.

Durante a execução do Filtro EKF e EKS de segunda ordem, foi notado as mesmas características do filtro de primeira ordem, a determinação das velocidades do objeto após a realização das mudanças de direção tem se mostrado com performance inferior a determinação da posição do objeto pelos Filtros visualizado na Figura 6-38.



**Figura 6-38 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.1).**

Os Filtros UKF e UKS tiveram uma baixa performance na determinação da posição e velocidade do objeto durante metade do trajeto proposto como exibido na Figura 6-38.



**Figura 6-39 - Posicionamento utilizando o UKF e URTS (Trajetória 3 e Pos.1).**

No geral pode ser notado na Tabela 6-12 o desempenho similar dos Filtros, com uma leve queda de performance do filtro EKF de segunda ordem.

**Tabela 6-12 - Erros Posicionamento (Trajetória 3 e Pos.1)**

<b>Filtros</b>	<b>Erro (RMSE)</b>
<b>EKF</b>	<b>0,0756</b>
<b>ERTS</b>	<b>0,0518</b>
<b>EKF 2</b>	0,0775
<b>ERTS 2</b>	0,0605
<b>UKF</b>	<b>0,0785</b>
<b>URTS</b>	<b>0,0519</b>

• **Trajétoria 3 Posição 2**

A mudança de posição agravou os erros no ajuste do modelo ao trajeto simulado, durante a fase inicial do percurso (Figura 6-39 e Figura 6-40), a difícil adaptação do Filtro EKF e EKS durante esta parte da trajetória.

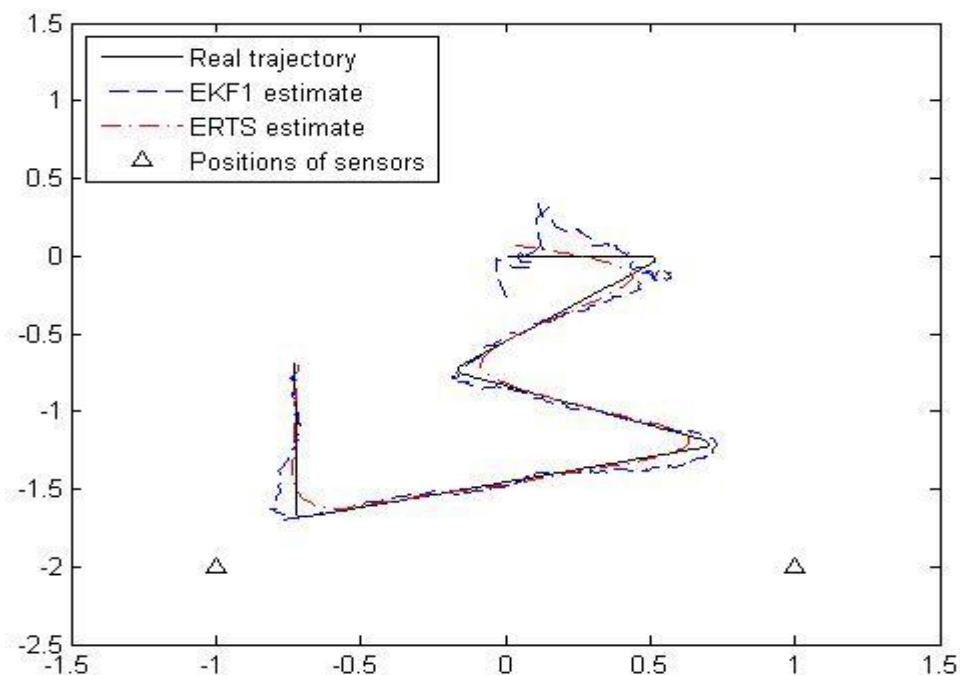
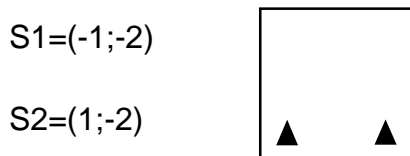


Figura 6-40 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.2).

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:



Porém no geral houve boa generalização e adaptação do modelo, tornando cada vez mais clara a dependência do modelo em relação a posição espacial dos sensores.

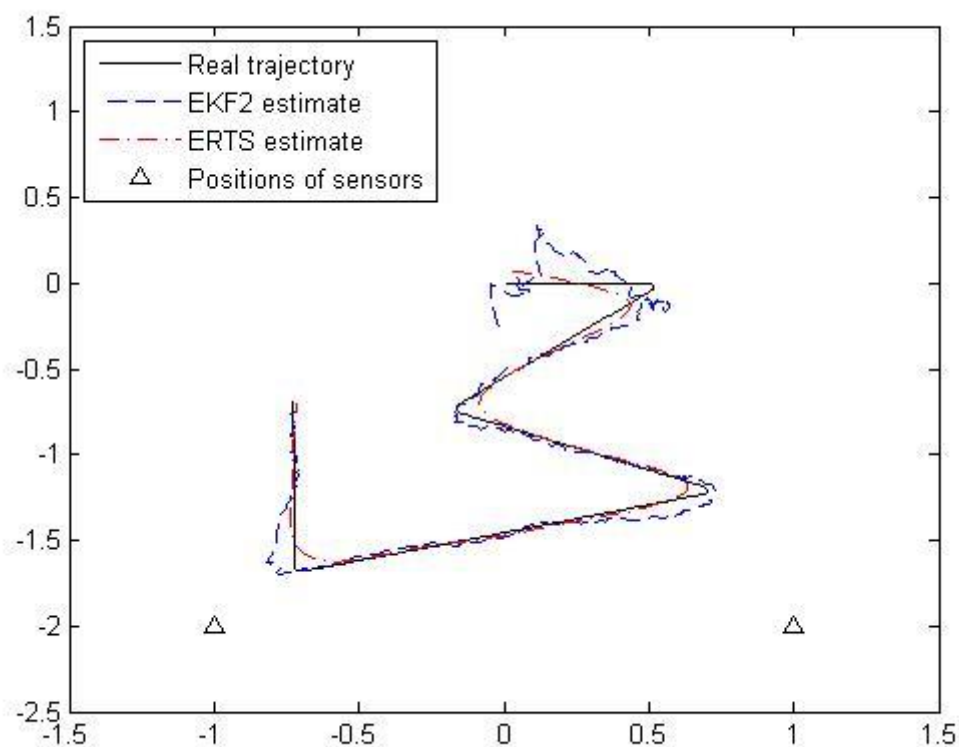


Figura 6-41 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.2).

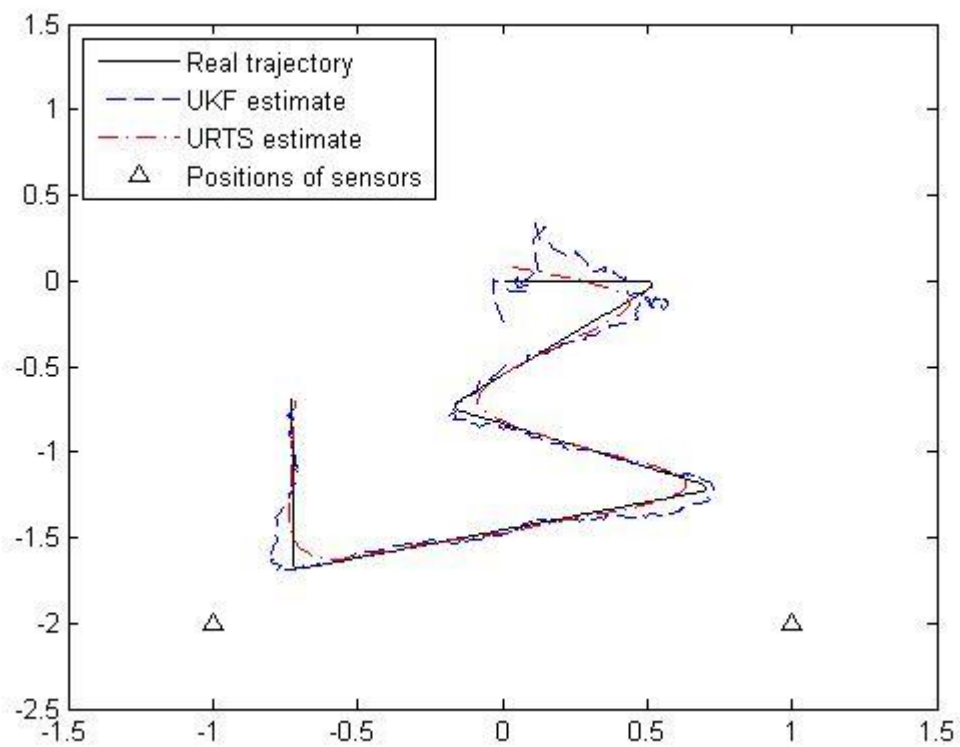


Figura 6-42 - Posicionamento utilizando o UKF e URTS (Trajetória 3 e Pos.2).

A dificuldade inicial em estimar o percurso inicial durante a utilização dos sensores na Posição 2, tem relação direta a posição espacial dos mesmos, deve ser analisado em um estudo futuro a possibilidade em determinar a melhor posição para os sensores de rolamento.

Durante boa parte do estudo o Filtro UKF (Figura 5-41) mostrou-se com maior acurácia em comparação ao Filtro EKF, porém na utilização do trajeto com mudanças repentinas de direção o filtro EKF obteve desempenho melhor ou similar ao UKF.

A Tabela 6-13 representa claramente essa similaridade de performance entre os Filtros testados.

**Tabela 6-13 - Erros Posicionamento (Trajetória 3 e Pos.2)**

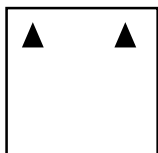
<b>Filtros</b>	<b>Erro (RMSE)</b>
<b>EKF</b>	0,0778
<b>ERTS</b>	0,0367
<b>EKF 2</b>	0,0776
<b>ERTS 2</b>	0,0365
<b>UKF</b>	0,0773
<b>URTS</b>	0,0368

- **Trajétoria 3 Posição 3**

Com os sensores localizados na Posição 3 como na Figura 6-43, um fato interessante ocorre, a dificuldade na determinação da velocidade não ocorre mais no início do trajeto, evidenciando a clara dependência do modelo a posição dos sensores.

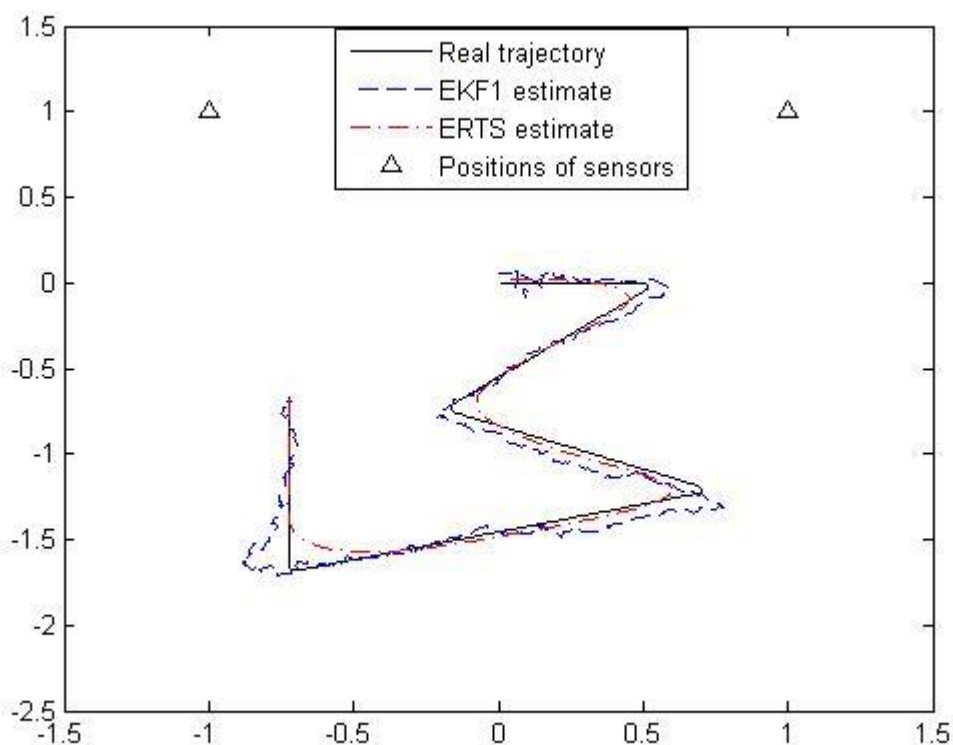
Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

S1=(-1;1)



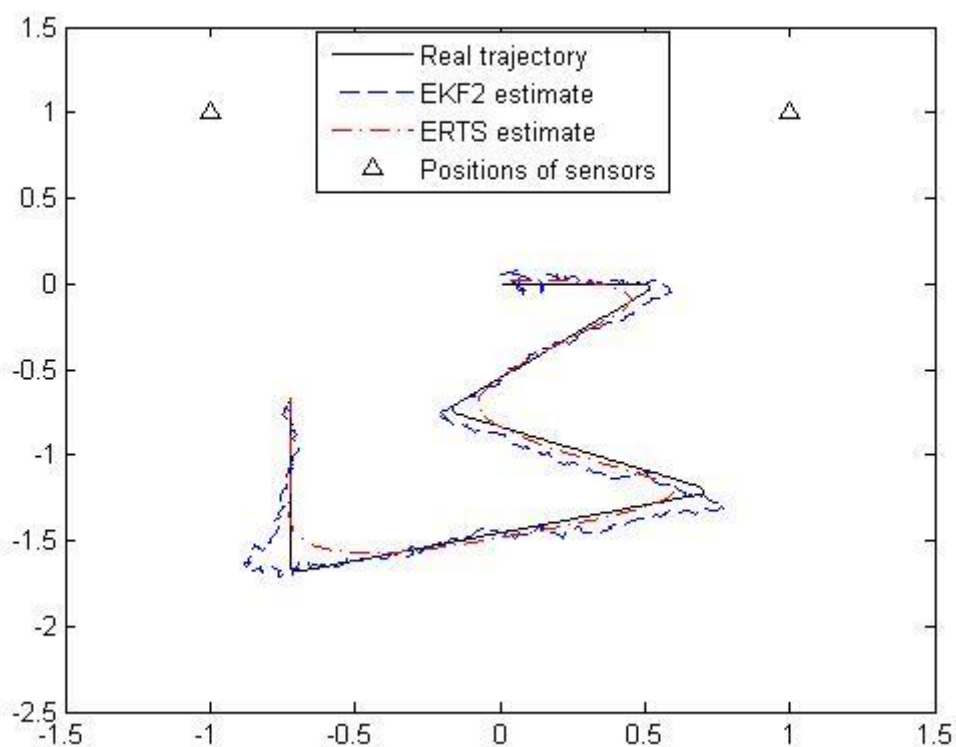
S2=(1;1)

Sendo esta uma lacuna a ser estudada sobre o método de estruturação ambiental utilizando sensores de rolamento, em um futuro estudo pode ser viabilizado uma forma analítica ou empírica de determinar a melhor posição dos sensores sobre o plano de trabalho em função a trajetória percorrida.



**Figura 6-43 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.3).**

Nas figuras 6-43 e 5-44 que demonstram a execução dos Filtros EKF e EKS de primeira e segunda ordem, sendo exibido o padrão de previsão dos mesmos sobre o trajeto, a determinação de posição e velocidade do objeto tem um aumento do erro no fim do percurso, erro este associado claramente a posição espacial dos sensores.



**Figura 6-44 - Posicionamento utilizando o EKF e ERTS de 2º ordem (Trajetória 3 e Pos.3).**

Pensando em casos reais durante a exploração do fundo do mar por sensores ultrassônicos ou efeito *Doppler*, o correto posicionamento dos sensores será fator preponderante no bom desempenho do estudo utilizando o método Bearing Only descrito em Bar-Shalom, Li e Kirubarajan (2001).

Durante a execução do Filtro UKF e UKS (Figura 6-45), ocorre uma situação interessante o Filtro tem uma baixa performance na previsão sobre o trajeto simulado.

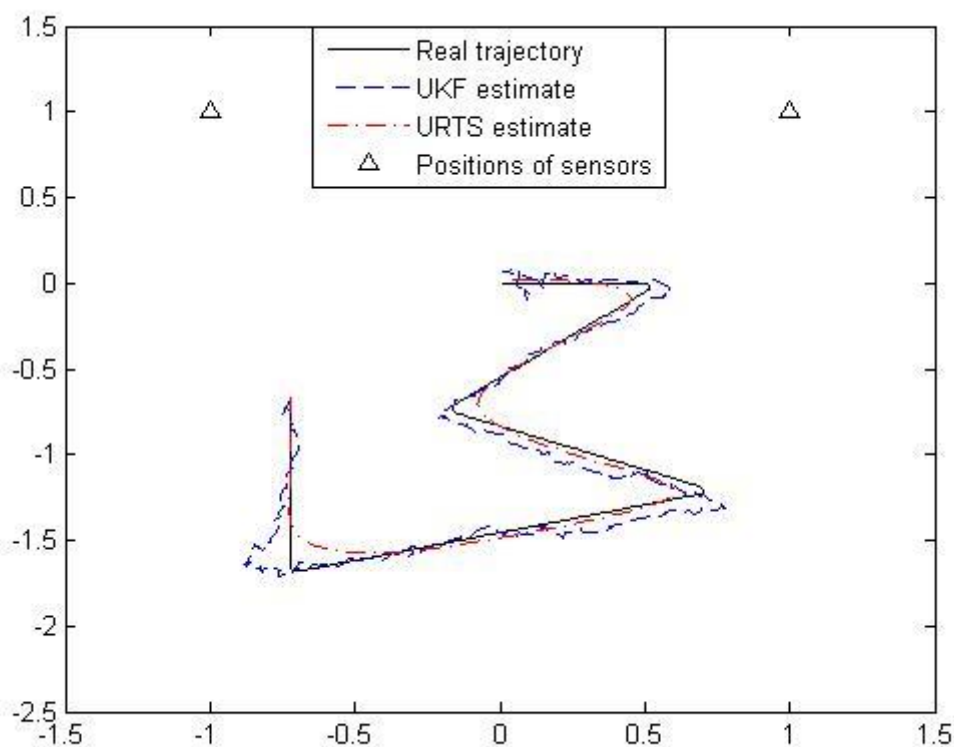


Figura 6-45 - Posicionamento utilizando o UKF e UKS (Trajetória 3 e Pos.3).

Fato este que não ocorreu durante as simulações executadas anteriormente como exibido na Tabela 6-14, a dependência do modelo em relação a posição espacial dos sensores e trajetória simulada são fatores preponderantes nesta ocorrência.

Tabela 6-14 - Erros Posicionamento (Trajetória 3 e Pos.3)

Filtros	Erro (RMSE)
EKF	0,0890
ERTS	0,0484
EKF 2	0,0890
ERTS 2	0,0483
<b>UKF</b>	<b>0,0910</b>
<b>URTS</b>	<b>0,0505</b>



- **Trajétoria 3 Posição 4**

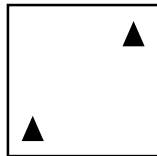
O Filtro EKF e EKS de primeira ordem obtiveram a pior performance durante todo o estudo na Posição 4 em relação aos sensores.

A disposição diagonal dos sensores em relação ao percurso proposto em todas a simulação foi fator agravante no aumento do erro.

Os sensores estão posicionados no plano cartesiano de trabalho simulado em:

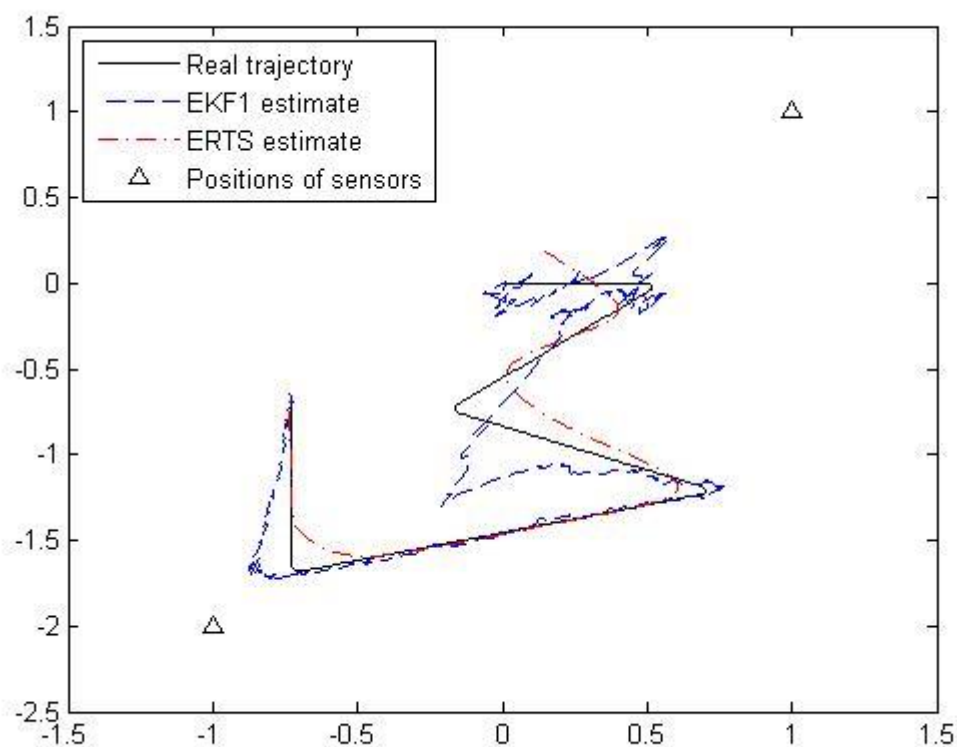
S1=(-1;-2)

S2=(1;1)



A dificuldade na predição da velocidade e posição do objeto durante as últimas simulações do método comprova a dependência da posição escalar dos sensores em relação a trajetória e/ou percurso a ser realizado por determinado objeto.

Podendo ser este um novo campo de estudo afim criar uma heurística de busca na solução otimizada para esse problema de rastreamento conforme mostrado na Figura 6-46.



**Figura 6-46 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.4).**

Sendo apontado nas Figuras 5-47 e 5-48, a baixa performance de todos os filtros, foi um fato surpreendente durante o estudo, fato este que não ocorreu durante as simulações iniciais.

O aumento dos erros ocorridos durante esta execução do modelo utilizando todos os métodos de filtragem propostos, exibem a fragilidade do modelo Bearing Only quanto a determinação errônea da posição dos sensores pelo experimentador, podendo ser este fator determinante entre o sucesso ou fracasso da pesquisa, sendo representativo esse fato nas Figuras 6-46, 6-47, 6-48.

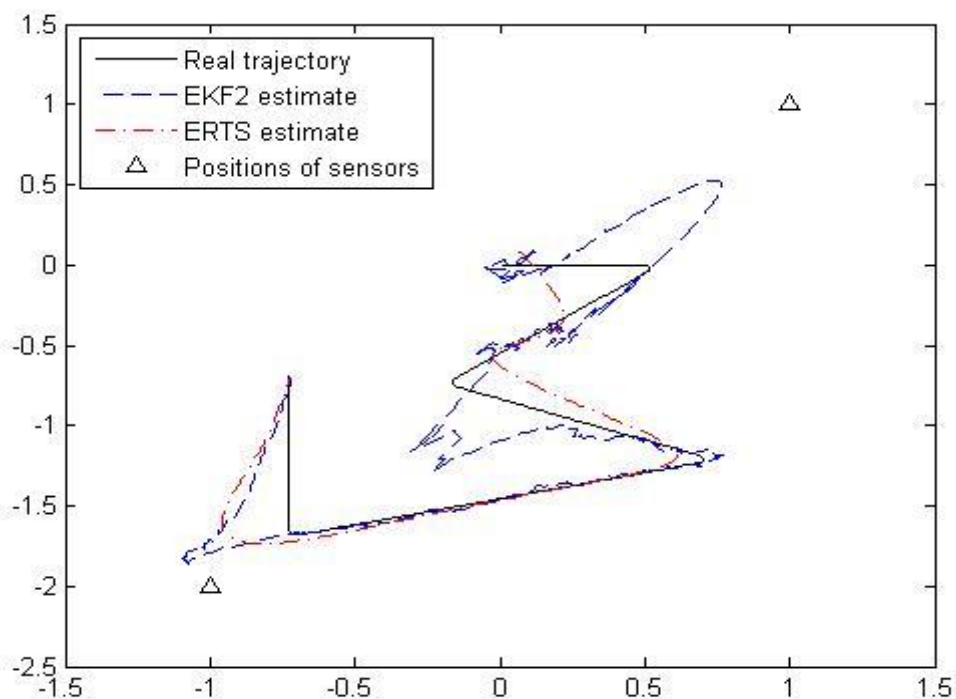


Figura 6-47 - Posicionamento utilizando o EKF e EKS de 2º ordem (Trajetória 3 e Pos.4).

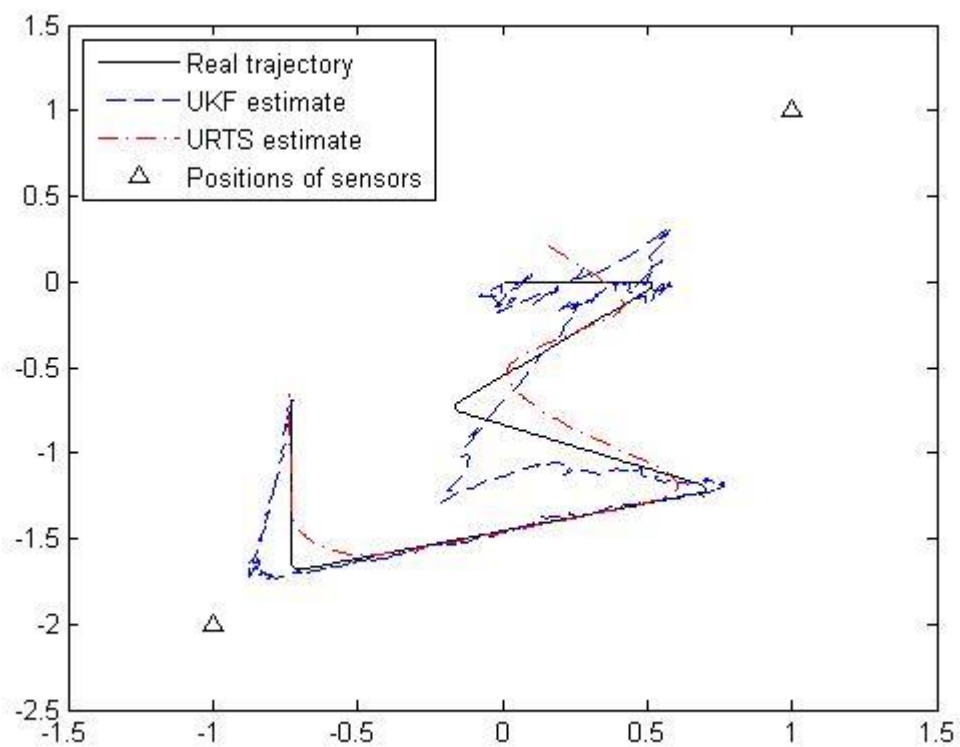


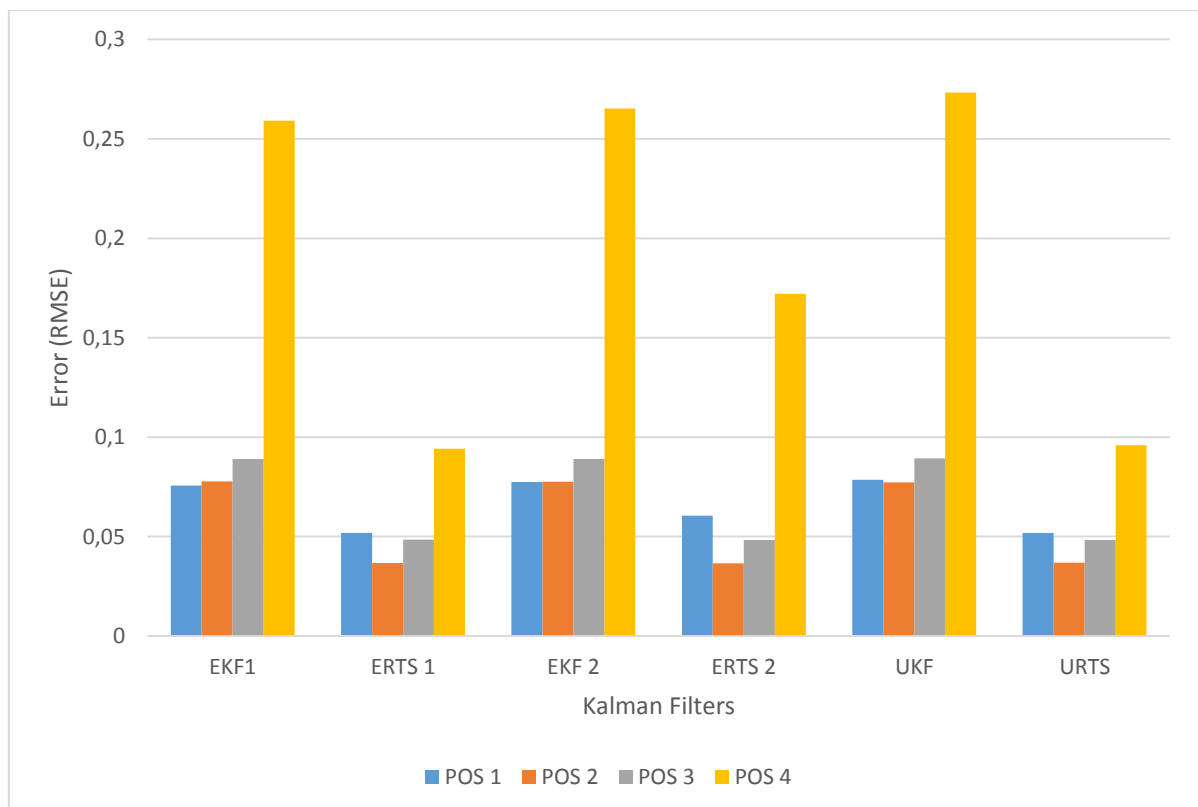
Figura 6-48 - Posicionamento utilizando o EKF e ERTS de 1º ordem (Trajetória 3 e Pos.4).

A Tabela 6-15 demonstra a menor eficácia dos filtros na predição dessa trajetória, a mudança repentina de trajetória e a disposição dos sensores tem papel fundamental nesse baixo desempenho.

**Tabela 6-15 - Erros Posicionamento (Trajetória 3 e Pos.4)**

Filtros	Erro (RMSE)
EKF	0,2591
ERTS	0,0942
EKF 2	0,2653
ERTS 2	0,1721
UKF	0,2733
URTS	0,0960

Na figura 6-49 fica evidente que a posição 4 referente aos sensores oferece um ambiente propício a erros e pouco favorável a correta predição do sistema.



**Figura 6-49 - Comparação entre a Posição dos Sensores x Erros (Trajetória 3).**

## 6.4 Considerações Finais

Durante os resultados de nossa pesquisa ficou evidente a dependência do modelo quando a disposição espacial dos sensores angulares, os autores estudados ao longo do trabalho não descreveram esse problema.

Nota-se pelas Figuras 6-46,6-47,6-48 e 6-49, o aumento do erro que ocorre quando a simulação é realizada utilizando os sensores na posição 4, sendo demonstrado nos três trajetos proposto.

Esses resultados podem ser de grande valia na determinação da posição espacial dos sensores, cabendo aos pesquisadores e desenvolvedores abordar novas estratégias para determinação destas posições.

Especialmente no início do deslocamento nas trajetórias propostas ocorre grande dificuldade em determinar a real posição, isso é causado pela grande incerteza em relação a velocidade inicial, após o deslocamento inicial os algoritmos obtêm respostas similares.

# Capítulo 7

## CONCLUSÃO

---

---

Os assuntos abordados nesse trabalho estão divididos em três pesquisas distintas e complementares, Filtros de Kalman e sua performance na previsão de series, Filtros de Kalman para o treinamento de Redes Neurais e Filtros de Kalman para determinação de Velocidade e Posição de Objetos utilizando a metodologia Bearing Only.

O primeiro trabalho baseado em Filtros de Kalman e sua efetividade na predição de series, obtendo resultados promissores e norteando trabalhos posteriores.

O trabalho comparou algumas das modificações do Filtro de Kalman, todas com importância em diversos campos do conhecimento e com seu estado da arte bem definido na literatura acadêmica.

Os bom desempenho obtido durante as tarefas executadas em outros trabalhos relacionados foram das variantes EKF, CKF e UKF necessitando de uma comparação para evidenciar potenciais características positivas de cada método, obtendo melhor desempenho na tarefa proposta em nosso trabalho a variante UKF.

Os resultados demonstraram o desempenho de cada variante, com ênfase no desempenho dos Filtros EKF e UKF, sendo utilizado posteriormente em nosso trabalho os Filtros EKF, UKF e suas variante de ordem maior e/ou suavizadas.

A avaliação do desempenho se fez necessária para ratificar trabalhos anteriores e demonstrar aos leitores a viabilidade do trabalho proposto.

Durante a segunda parte do trabalho foi implementado/analísado, qualitativamente e quantitativamente duas metodologias de treinamento de Redes Neurais utilizando os Filtros UKF e EKF como metodologia.

O resultado obtido mostrou a viabilidade de se utilizar os métodos e suas características, relacionadas ao tempo de treinamento, quantidade de iterações necessárias para aprendizado, curva de desempenho e erro associado.

A maior contribuição desse segundo trabalho foi demonstrar de forma empírica, uma forma de determinar a quantidade de neurônios na camada oculta que resultaria em um melhor desempenho no treinamento, fato este não analisado pelos autores citados acima e desenvolvedores dos métodos.

Os resultados desse trabalho fomentaram a ideia de trabalhos futuros que possam utilizar o Filtro de Kalman e Redes Neurais de forma cooperativa, paralela e/ou corretiva durante tarefas de posicionamento, rastreamento e predição.

Com os resultados das pesquisas efetuadas anteriormente e as lições aprendidas, foi realizado o trabalho com maior importância em nossa pesquisa e tema central de nossos esforços.

Sendo este trabalho tema de grande discussão entre pesquisadores, pois a metodologia *Bearing Only* pode ser utilizada para posicionamento e determinação de posição e velocidade em objetos localizados em ambientes não estruturados, com a adição de Sensores (Doppler, Ultrassom, Lasers), podendo ser utilizado em ambientes aquáticos, industriais, aéreos e espaciais, sendo este uma de suas principais vantagens.

Foram utilizados movimentos baseados em deslocamento com acelerações ao longo do percurso, sendo eles Deslocamento Linear com curvas Suaves, Deslocamento com perfil Circular e Deslocamento com mudanças bruscas de Direção

Tendo como característica da avaliação utilizar posições espaciais, pré-definidas para os sensores de rolamento, tentando obter parâmetro empírico que justifique este posicionamento dos sensores no plano pelo experimentador.

Os resultados demonstraram que no deslocamento Linear com Curvas suaves obteve boa generalização dos algoritmos quanto a posição e velocidade do objeto em estudo, e forte influência do posicionamento dos sensores no plano comparado a razão da performance dos algoritmos na predição do sistema, tendo a Posição 4 dos sensores o pior desempenho na tarefa de predição proposta para o Trajeto 1.

A trajetória com percurso circular obteve um rendimento menor por parte dos algoritmos, as acelerações iniciais e durante o percurso trazem dificuldade do algoritmo de prever o movimento, aliado à disposição espacial dos sensores que tende a elevar a complexidade da percepção espacial do modelo.

Porém o sistema obteve bons resultados sendo necessário estudos mais aprofundados no cerne do método a fim de identificar o viés que causa essas derivas, podendo ser fonte de estudo em trabalhos futuros, a trajetória completamente circular tem pouco impacto em rastreamento de objetos em sistemas de deslocamento: Industrial (Esteiras Transportadoras) e Aquáticos (Sistemas de Soldagem em Águas Profundas).

O rastreamento de objetos com variações bruscas durante o movimento, tem um papel importante nesse trabalho, pois tem foco nas tarefas de posicionamento, rastreamento, detecção, criação de mapas e estruturação ambiental como um todo.

Os algoritmos e métodos atingiram excelentes resultados, comprovando a real dependência do modelo em relação a disposição espacial dos sensores no plano de trabalho simulado e trouxeram à tona um problema a ser estudado em sua extensão.

Todas as metodologias e algoritmos foram eficientes em suas tarefas e cada qual com sua característica acentuada para determinados trajetos, localização espacial e a determinação de velocidade e posição do objeto.

Portanto pode concluir que a posição dos sensores no plano de simulação tem um papel fundamental no desempenho dos algoritmos sendo fator decisivo no sucesso da aplicação seja ela simulada ou aplicada em ambientes reais, cabendo ao



desenvolvedor intensa pesquisa desse ponto crucial, tendo o ambiente simulado grande valia neste processo.

Os algoritmos e métodos demonstraram a possibilidade de serem aplicados em quaisquer tipos de movimentação relacionado aos ambientes acadêmicos e industrias respeitando os limites de deslocamento sobre o plano XY.

Baseado nos resultados obtidos pela nossa simulação, é observado que o método pode ser aplicado em ambientes reais (sendo necessárias testes em ambientes reais com sensores de forma controlada), podendo ser avaliado deste modo, ruídos e outras formas de distorção dos sinais que não foram avaliadas durante as simulações.

## 7.1 Trabalhos Futuros

Algumas pesquisas e propostas de trabalhos futuros podem ser realizadas a partir do método proposto, podendo ser:

- Implementação do Método Neuro-Estatístico com utilizando as redes Elman (ELMAN, 1990), conforme descrito em (ZHONG, ZHONG e PENG, 2013), podendo ser testado esse Filtro de Kalman assistido e/ou associado as Redes em Elman em sistemas de predição on-line, portanto sem a necessidade de treinamento inicial para conhecimento do modelo ou sistema proposto.
- Implementação do método proposto em um ambiente real controlado.
- Implementar o método proposto em *hardware de alto desempenho*.
- Desenvolvimento de um sistema conjunto utilizando Redes Neurais e Filtros de Kalman para o posicionamento utilizando a metodologia Bearing Only.
- Avaliar os métodos propostos em sistemas que se utilizam de sensores de rolamento (Ultrassom, Doppler, Laser) para posicionamento em ambientes de difícil percepção espacial e/ou não estruturados.

# REFERÊNCIAS

---

---

ARASARATNAM, I.; HAYKIN, S. Nonlinear Bayesian Filters for Training Recurrent Neural Networks. **MICAI 2008: Advances in Artificial Intelligence**, 2008. 12-33.

ARASARATNAM, I.; SIMON, H. Cubature Kalman Filters. **IEEE Transactions on Automatic Control**, v. 54, p. 1254-1269, June 2009.

AUFRKRE, R. et al. **Road sides recognition in non-structured environments by vision**. 2004 IEEE Intelligent Vehicles Symposium. Parma, Italy: [s.n.]. 2004. p. 329-334.

AUGER, F. et al. Industrial Applications of the Kalman Filter: A Review. **IEEE Transactions on Industrial Electronics**, 60, Dec. 2013. 5458-5471.

BALDI, P. F.; HORNIK, K. Learning in linear neural networks: a survey. **IEEE Transactions on Neural Networks**, 6, Jul 1995. 837-858.

BAR-SHALOM, Y.; LI, X. R.; KIRUBARAJAN, T. **Estimation with Applications To Tracking and Navigation: Theory Algorithms and Software**. [S.l.]: John Wiley & Sons, 2001. 584 p.

BISHOP, G. W. A. G. **An Introduction to the Kalman Filter**. Chapel Hill. 2006.

CHAUMETTE, F.; HUTCHINSON, S. Visual servo control, Part I: Basic approaches. **IEEE Robotics and Automation Magazine**, Dezembro 2006. 82-90.

---

CHAUMETTE, F.; HUTCHINSON, S. Visual servo control. Part II: Advanced approaches [Tutorial]. **Robotics & Automation Magazine, IEEE**, 14, n. 1, Março 2007. 109-118.

CHEN, S. Y. Kalman Filter for Robot Vision: A Survey. **IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL 59, NO. 11**, NOVEMBER 2012.

CORKE, P. I. **Robotics, Vision and Control: Fundamental Algorithms in MATLAB**. [S.I.]: Springer, v. 1, 2011. 572 p.

CORKE, P. I.; GOOD, M. C. Controller design for high-performance visual servoing. **Proc. 12th World Congr. IFAC'93**, Sydney, Australia, Julho 1993. 395-398.

CRAIG, J. J. **Introduction to Robotics: Mechanics and Control**. Upper Saddle River, NJ 07458: Prentice Hall, 2005.

CROUSE, D. F. One can do better than the unscented Kalman filter for multistatic tracking. **IEEE Aerospace Conference**, 2013. 1-20.

D'ALFONSO, L. et al. Filters for mobile robots: EKF, UKF and sensor switching - experimental results. **9th IEEE International Conference on Control and Automation (ICCA)**, 2011. 925-930.

DOLLAR, A. M.; HOWE, R. D. **Simple, Robust Autonomous Grasping in Unstructured**. IEEE International Conference on. Roma,Italy: [s.n.]. 2007. p. 4693-4700.

ELMAN, J. L. Finding structure in time. **Cognitive Science**, 14, n. 2, 1990. 179-211.

GREWAL, M. S.; ANDREWS, A. P. **Kalman Filtering**. New Jersey: Prentice-Hall,Inc, 1993.

GROOVER, M. P. **Automação Industrial E Sistemas de Manufatura**. 3. ed. [S.I.]: Pearson, 2010. 592 p.

H. E. RAUCH; STRIEBEL, C. T.; TUNG, F. Maximum likelihood estimates of linear dynamic systems. **AIAA Journal**, 3, n. 8, 1965. 1445-1450.

---

HARTIKAINEN, J.; SÄRKKÄ, S. **Optimal filtering with Kalman Filters and Smoothers – a Manual for Matlab Toolbox EKF/UKF**. Helsinki University of Technology: Department of Biomedical Engineering and Computational Science, 2008.

HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. [S.I.]: Prentice Hall, 1999. 700 p.

HAYKIN, S. **Kalman Filters in Kalman Filtering and Neural Networks**. Ontario, Canada: John Wiley and Sons, Inc., 2001. 1-22 p.

HROMKOVIČ, J. **Algorithms for hard problems: Introduction to combinatorial optimization, randomization, approximation, and heuristics**. Berlin: Springer-Verlag Berlin Heidelberg, 2004.

HSIAO, W.-L. C. A. F.-B. Implementation of the Rauch-Tung-Striebel Smoother for Sensor Compatibility Correction of a Fixed-Wing Unmanned. **Sensors**, 2011. 3738-3764.

HUTCHINSON, S.; HAGER, G. D.; CORKE, P. I. A tutorial on visual servo control. **Robotics and Automation, IEEE Transactions on**, 12, n. 5, Outubro 1996. 651-670.

IIGUNI, Y. A. S. H. A. T. H. A real-time learning algorithm for a multilayered neural network based on the extended Kalman filter. **IEEE Transactions on Signal Processing**, 1992. 8.

IKEDA, K.; DAIDO, H.; OAKIMOTO. Multiple-valued Stationary State and its Instability of the Transmitted Light by a Ring Cavity System, Opt. Commun. **Optical Turbulence: Chaotic Behavior of Transmitted Light from a Ring Cavity**, **phys. Rev. Lett.**, 1980. 709-712.

INFOESCOLA, 12 jun. 2015. Disponível em: <[www.infoescola.com.br](http://www.infoescola.com.br)>.

JACKSON, J. D. **CLASSICAL ELECTRODYNAMICS**. [S.I.]: John Wiley, 1999.

---

JANABI-SHARIFI, F.; MAREY, M. A Kalman-Filter-Based Method for Pose Estimation in Visual Servoing. **IEEE TRANSACTIONS ON ROBOTICS**, VOL 26, OCTOBER 2010.

JEONG-GWAN KANG, S.-Y. O. A. A. S.-Y. O. **Modified Neural Network aided EKF based SLAM for improving an accuracy of the feature map**. The 2010 International Joint Conference on Neural Networks (IJCNN). [S.I.]: [s.n.]. 2010. p. 1-7.

JIA, B.; XIN, M.; CHENG, Y. The high-degree cubature Kalman filter. **IEEE 51st Annual Conference on Decision and Control (CDC)**, 2012. 4095-4100.

JULIER, S. J.; UHLMANN, J. K. **A New Extension of the Kalman Filter to Nonlinear Systems**. The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Multi Sensor Fusion, Tracking and Resource Management 11, SPIE. [S.I.]: [s.n.]. 1997.

K.RADHAKRISHNAN; UNNIKRISHNAN, A.; BALAKRISHNAN, K. G. **Bearing only Tracking of Maneuvering Targets using a single coordinated turn model**. International Journal of Computer Applications. [S.I.]: [s.n.]. 2010. p. 25-33.

KALMAN, R. E. A New Approach to Linear Filtering and Prediction. **Transaction of the ASME—Journal of Basic Engineering**, p. 35-45, 1960.

KHAN, M. W. et al. **A comparative study of target tracking with Kalman filter, extended Kalman filter and particle filter using received signal strength measurements**. Emerging Technologies (ICET), 2015 International Conference on. [S.I.]: [s.n.]. 2015. p. 1-6.

KOREN, Y. **Robotics for Engineers**. [S.I.]: McGraw Hill, 1985.

KOTTECHA, J. H.; DJURIC, P. M. Gaussian particle filtering. **IEEE Transactions on Signal Processing**, 51, 2003. 2592-2601.

KURT-YAVUZ, Z.; YAVUZ, S. A comparison of EKF, UKF, FastSLAM2.0, and UKF-based FastSLAM algorithms. **IEEE 16th International Conference on Intelligent Engineering Systems (INES)**, June 2012. 37-43.

---

LAGADIC, I. **VISP. VISP**, 2013. Disponível em: <<http://www.irisa.fr/lagadic/visp/visp.html>>. Acesso em: Dezembro 2013.

LAVIOLA, J. J. **A comparison of unscented and extended Kalman filtering for estimating quaternion motion**. Proceedings of American Control Conference. [S.l.]: [s.n.]. 4-6 June 2013. p. 2435-2440.

LEHMANN, E. L.; CASELLA, G. **Theory of Point Estimation (2nd ed.)**. New York: Springer, 1998.

LIU, C.; SHUI, P.; LI, S. Unscented extended Kalman filter for target tracking. **Journal of Systems Engineering and Electronics**, 22, April 2011. 188-192.

LORENZ, E. The Statistical Prediction of Solutions of Dynamic Equations. **Symposium on Numerical Weather Prediction in Tokio**, 1960.

MALLICK, M.; MORELANDE, M.; MIHAYLOVA, L. Continuous-discrete filtering using EKF, UKF, and PF. **15th International Conference on Information Fusion (FUSION)**, 2012. 1087-1094.

MAYBECK, P. **Stochastic Models, Estimation and Control**. [S.l.]: Academic Press, v. 141, 1982.

MITCHEL, T. **Machine learning: An artificial intelligence approach**. [S.l.]: McGraw Hill, 1997.

MONTESANO, L.; MINGUEZ, J.; MONTANO, L. **Probabilistic scan matching for motion estimation in unstructured environments**. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005). [S.l.]: [s.n.]. 2005. p. 3499-3504.

NEMEC, B.; ZLAJPAH, L. Force Control of Redundant Robots in Unstructured Environments. **IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 49, NO. 1, FEBRUARY 2002**, 2002. 233-240.

PAZOS, F. **Automação de Sistemas e Robótica**. Rio de Janeiro: Axcel Books, 2002.

---

RAMAKOTI, N.; VINAY, A.; JATOTH, R. K. **Particle Swarm Optimization Aided Kalman Filter for Object Tracking**. International Conference on Advances in Computing, Control, and Telecommunication Technologies. [S.I.]: IEEE Computer Society. 2009. p. 531-533.

ROMANO, V. F.; FERREIRA, C. V. Introdução à Robótica Industrial. In: ROMANO, V. **Robótica Industrial: Aplicação na Indústria de Manufatura e de Processos**. São Paulo: Edgard Blucher, 2002. p. 1-19.

SÄRKKÄ, S. On Unscented Kalman Filtering for State Estimation. **IEEE TRANSACTIONS ON AUTOMATIC CONTROL**, 2007.

SÄRKKÄ, S.; VEHTARI, A.; LAMPINEN, J. Rao-Blackwellized particle filter for multiple target tracking. **Seventh International Conference on Information Fusion (FUSION 2004)**, 2007.

SCHEERING, C.; ZHANG, J.; KNOLL, A. **Distributed Multiple View Fusion for Two-Arm Distance Estimation**. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems. Taipei, Taiwan: [s.n.]. 1999. p. 38-43.

SILVA, L. F. P. C. E. Utilização de Manipuladores em Ambientes Não Estruturados. **Dissertação de Mestrado**, 23 Julho 2010. 99.

SIMO SÄRKKÄ, A. V. L. Probabilistic Methods in Multiple Target Tracking (Review and Bibliography). **Research report B36. ISBN 951-22-6938-4.**, 2004.

STUBBERUD, S. C.; LOBBIA, R. N.; OWEN, M. An adaptive extended Kalman filter using artificial neural networks, 2, 13-15 DEC 1995. 1852-1856.

STUBBERUD, S. C.; OWEN, M. W. Targeted on-line modeling for an extended Kalman filter using artificial neural networks, 2, 1998. 1019-1023.

SUN, M.; MA, Z.; LI, Y. **Maneuvering Target Tracking Using IMM Kalman Filter Aided by Elman Neural Network**. Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2015 7th International Conference on. [S.I.]: [s.n.]. 2015. p. 144-148.



---

TATIBANA, C. Y. . K. D. Y. . A. A. R. N. . A. C. D. R. N. . D. R. A. . &. D. A. D. Seja bem vindo a homepage de Redes Neurais, 2004.

TORRES, J. P. A. F. Movement-Flow-Based Visual Servoing. **IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 35, NO. 1**, p. 4-15, 2005.

VATAVU, A.; NEDEVSKI, S. Vision-based Tracking of Multiple Objects in Dynamic Unstructured Environments Using Free-Form Delimiters. **European Conference of Mobile Robots**, 2013. 367-372.

WAN, E. A.; VAN DER MERWE, R. **The unscented Kalman filter for nonlinear estimation**. Adaptive Systems for Signal Processing, Communications, and Control Symposium. [S.l.]: [s.n.]. 2000. p. 153-158.

WAN, R. Z. A. J. Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation. **Signal Processing Letters, IEEE**, 2006. 6.

WANG, X.; DENG, Z. **Robust centralized fusion steady-state Kalman filter for multisensor uncertain systems**. Estimation, Detection and Information Fusion (ICEDIF), 2015 International Conference on. [S.l.]: [s.n.]. 2015. p. 100-104.

WARSOP, T. A. S. S. **A survey of object recognition methods for automatic asset detection in high-definition video**. Cybernetic Intelligent Systems (CIS), 2010 IEEE 9th International Conference on. [S.l.]: [s.n.]. 2010. p. 1-6.

WECKESSE, P.; DILLMANN, R. Modeling unknown environments with a mobile robot. **Robotics and Autonomous Systems 23**, p. 293-300, 1998.

WIENER, N. **Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications**. [S.l.]: MIT Press, 1949.

WU, Y. et al. A Numerical-Integration Perspective on Gaussian Filters. **IEEE Transactions on Signal Processing**, 2006. 2910-2921.

ZHAN, R.; WAN, J. Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation. **IEEE Signal Processing Letters**, 13, July 2006. 445-448.

ZHANG, J.; SCHMIDT, R.; KNOLL, A. **Appearance-based visual learning in a neuro-fuzzy model for fine-positioning of manipulators**. IEEE International Conference on Robotics and Automation. [S.l.]: [s.n.]. 1999. p. 1164-1169.

ZHONG, X.; ZHONG, X.; PENG, X. Robust Kalman Filtering Cooperated Elman Neural Network Learning for Vision-Sensing-Based Robotic Manipulation with Global Stability. **Sensors (Basel)**, p. 13464-13486, 2013.

ZUBEN, L. N. D. C. & F. J. V. Immune and Neural Network Models:Theoretical and Empirical Comparisons. **International Journal of Computational Intelligence and Applications (IJCIA)**, p. 239-257, 2001.