



**UNIVERSIDADE FEDERAL DO CEARÁ  
CENTRO DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**THIAGO DINIZ LOBO**

**CONTROLE ADAPTATIVO AUTO-AJUSTÁVEL PARA CONTROLE DE  
TRAJETÓRIAS DE UM ROBÔ MÓVEL COM RODAS**

**FORTALEZA**

**2016**

THIAGO DINIZ LOBO

CONTROLE ADAPTATIVO AUTO-AJUSTÁVEL PARA CONTROLE DE  
TRAJETÓRIAS DE UM ROBÔ MÓVEL COM RODAS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de concentração: Sistemas de Energia Elétrica.

Orientador: Prof. Dr. Fabrício Gonzalez Nogueira.

Coorientador: Prof. Dr. Bismark Claure Torrico.

FORTALEZA

2016

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- L785c Lobo, Thiago Diniz.  
Controle adaptativo auto-ajustável para controle de trajetórias de um robô móvel com rodas / Thiago Diniz Lobo. – 2016.  
74 f. : il. color.
- Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Fortaleza, 2016.  
Orientação: Prof. Dr. Fabrício Gonzalez Nogueira.  
Coorientação: Prof. Dr. Bismark Claire Torrico.
1. Controle Adaptativo. 2. Robô Móvel com Duas Rodas. 3. Seguimento de Trajetórias. I. Título.  
CDD 621.3
-

THIAGO DINIZ LOBO

CONTROLE ADAPTATIVO AUTO-AJUSTÁVEL PARA CONTROLE DE  
TRAJETÓRIAS DE UM ROBÔ MÓVEL COM RODAS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Engenharia Elétrica. Área de concentração: Sistemas de Energia Elétrica.

Aprovada em: \_\_\_/\_\_\_/\_\_\_\_\_.

BANCA EXAMINADORA

---

Prof. Dr. Fabrício Gonzalez Nogueira (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Bismark Claire Torrico (Co-orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Paulo Peixoto Praça  
Universidade Estadual do Ceará (UFC)

---

Prof. Dr. George André Pereira Thé  
Universidade Estadual do Ceará (UFC)

A Deus.

Aos meus pais, familiares, amigos e namorada.

## **AGRADECIMENTOS**

Agradeço também ao meu orientador Prof. Fabrício Gonzalez Nogueira pela dedicação, apoio e compreensão.

Aos professores participantes da banca examinadora Prof. Bismark, Prof. Paulo Praça e Prof. George pelo tempo, pelas valiosas colaborações e sugestões.

Aos professores da Pós-Graduação em Engenharia Elétrica da Universidade Federal do Ceará pelos conhecimentos transmitidos em sala e extra sala.

Aos colegas de laboratório Marcus Forte, Rigoberto Sousa, Vanier Andrade, Adriano Rodrigues, Eduardo Mapurunga, dentre outros amigos de engenharia não citados mas que também ajudaram com discussões, reflexões, críticas e sugestões recebidas.

E, por fim, mas não menos importantes, agradeço aos amigos e parentes pelos incentivos e compreensão nos períodos em que tive de me ausentar para me dedicar na conclusão desse trabalho.

*“Lex I: Corpus omne perseverare in statu suo quiescendi vel movendi uniformiter in directum, nisi quatenus a viribus impressis cogitur statum illum mutare*

(Tradução) Lei I: Todo corpo continua em seu estado de repouso ou de movimento uniforme em uma linha reta, a menos que seja forçado a mudar aquele estado por forças aplicadas sobre ele.” (Primeira Lei de Newton).

“Para que a humanidade evolua, é necessário que saia da zona de conforto. Ou seja, para sair do estado de repouso, devemos procurar forças, sejam elas encontradas na ciência ou na religião, contanto que nos levem a evoluir.” (Autor, inspirado na Lei I de Newton).

## RESUMO

Este trabalho apresenta a implementação e testes experimentais de uma estratégia de controle adaptativo para seguimento de trajetórias por robôs móveis com acionamento diferencial e restrições não-holonômicas. O sistema de controle proposto é composto por duas malhas de controle em cascata, na qual a primeira consiste em um controlador PID responsável pelo controle de velocidade das rodas de tração e a segunda é um controlador não linear que tem o objetivo de permitir ao robô o seguimento de uma trajetória de referência. Para garantir a convergência assintótica do robô mesmo quando sujeito a incertezas, foi proposta a aplicação de um controlador não linear adaptativo com parâmetros variantes no tempo. Esse tipo de controlador possibilita o robô de se ajustar a diferentes tipos de trajetórias de referência, além de contornar problemas relacionados com distúrbios e compensar incertezas dos parâmetros do sistema. O modelo cinemático do robô é utilizado para obter uma estimação da postura do robô, permitindo assim o cálculo do erro com relação à trajetória de referência. Resultados de simulações computacionais e testes experimentais em um robô móvel real mostram vantagens e desvantagens do controlador adaptativo proposto.

**Palavras-chave:** Controle Adaptativo; Robô Móvel com Duas Rodas; Seguimento de Trajetórias.



## **ABSTRACT**

This work presents the implementation and experimental tests of an adaptive control strategy for tracking path of wheeled mobile robots (WMR) with differential drive and nonholonomic constraints. The proposed control system consists of two control loops cascaded, which the first is a PID controller responsible for the speed control of the drive wheels, and the second is a nonlinear controller which has the purpose of allowing the robot tracking a reference trajectory. To ensure the asymptotic convergence of the robot even when subject to uncertainty, it was proposed to apply a nonlinear adaptive controller with time varying parameters. This kind of controller allows the robot to adjust to different reference trajectories, and overcome problems related to disturbances and uncertainties in system parameters. In this work, the robot kinematic model was used to estimate the robot posture and to calculate the error between the real and the desired trajectory. Results of computational simulations and experimental tests carried out on a real mobile robot show advantages and disadvantages of the proposed adaptive controller.

**Keywords:** Adaptive Control; Wheeled Mobile Robots; Path Tracking Control.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	12
<b>1.1</b>	<b>Exemplos de Robôs Móveis Comerciais de uso Industrial e Militar</b> .....	13
<b>1.2</b>	<b>Levantamento do Estado da Arte</b> .....	15
<b>1.3</b>	<b>Objetivos do Trabalho</b> .....	18
<b>1.3.1</b>	<i>Objetivo Principal</i> .....	19
<b>1.3.2</b>	<i>Objetivos Específicos</i> .....	19
<b>1.4</b>	<b>Estrutura da Dissertação</b> .....	19
<b>2</b>	<b>MODELAGEM DE ROBÔS MÓVEIS NÃO-HOLONÔMICOS</b> .....	20
<b>2.1</b>	<b>Introdução</b> .....	20
<b>2.2</b>	<b>Sistemas não-holonômicos</b> .....	21
<b>2.3</b>	<b>Modelagem Matemática de um Robô Móvel Não-holonômico</b> .....	22
<b>3</b>	<b>DESCRIÇÃO DO CONTROLADOR BACKSTEPPING ADAPTATIVO</b> .....	28
<b>3.1</b>	<b>Introdução</b> .....	28
<b>3.2</b>	<b>Controlador Não-linear para Rastreamento de Trajetórias</b> .....	28
<b>3.2.1</b>	<i>Geração de trajetória</i> .....	29
<b>3.2.2</b>	<i>Transformação de Frame</i> .....	29
<b>3.2.3</b>	<i>Controlador Backstepping</i> .....	29
<b>3.2.4</b>	<i>Dinâmica do robô</i> .....	30
<b>3.2.5</b>	<i>Cinemática inversa</i> .....	31
<b>3.3</b>	<b>Estratégia de adaptação online dos ganhos do controlador backstepping</b> .....	31
<b>3.4</b>	<b>Cálculo da Jacobiana usando as equações exatas do sistema</b> .....	36
<b>3.5</b>	<b>Discretização para implementação no robô real</b> .....	39
<b>3.6</b>	<b>Estratégias de controle não-linear baseada em (Klancar, Matko, &amp; Blazic, 2005)</b> .....	40

<b>4</b>	<b>RESULTADOS DE SIMULAÇÃO</b> .....	<b>42</b>
<b>4.1</b>	<b>Ajuste dos controladores</b> .....	<b>47</b>
<b>4.2</b>	<b>Avaliação do desempenho dos controladores para diferentes trajetórias</b> .....	<b>52</b>
<b>4.3</b>	<b>Avaliação do desempenho dos controladores contra perturbações e distúrbios</b>	<b>61</b>
<b>5</b>	<b>RESULTADOS EXPERIMENTAIS</b> .....	<b>65</b>
<b>5.1</b>	<b>Ensaio para trajetória de referência de um oito</b> .....	<b>65</b>
<b>6</b>	<b>CONCLUSÃO</b> .....	<b>71</b>
	<b>REFERÊNCIAS</b> .....	<b>72</b>

## 1 INTRODUÇÃO

Ao longo das últimas décadas a robótica tem despertado a imaginação e o interesse da humanidade. Utilizada em diversos ambientes (industrial, militar, comercial, hospitalar ou doméstico), a robótica móvel é amplamente aplicável em trabalhos repetitivos, insalubres e/ou perigosos ao ser humano, como em linhas de produção industriais, exploração de ambientes inóspitos, cirurgias, busca e desarmamento de bombas, dentre outras aplicações. Para explorar esse potencial, países líderes no ramo da tecnologia passaram a investir cada vez mais em pesquisa e desenvolvimento na área da robótica. Por exemplo, a Coreia do Sul investiu em dez anos, aproximadamente, 1 bilhão de dólares no desenvolvimento de tecnologias robóticas, e a União Europeia planejou investir 536 milhões de euros em pesquisas relacionadas às áreas de robótica entre os anos de 2007 e 2012 (ROMERO, PRESTES, *et al.*, 2014).

É comum encontrar na literatura a divisão dos robôs em dois grupos. O primeiro engloba os robôs manipuladores, amplamente utilizados no ambiente industrial, realizando tarefas repetitivas com alto grau de precisão, tais como em processos de soldagem, pintura, inspeção e corte (CRAIG, 2008). E no segundo grupo estão os robôs móveis, que possuem a capacidade de deslocar-se em ambientes terrestres, aéreos ou aquáticos (SIEGWART e NOURBAKHSI, 2004). Existem também robôs híbridos, isto é, compostos por manipuladores e plataformas móveis, como é o caso dos humanoides e veículos terrestres, aeroespaciais e submarinos dotados de braços manipuladores (VIEIRA, 2005).

Segundo Vieira (2005), a classe dos robôs móveis pode ser constituída pela capacidade de locomoção no espaço tridimensional (robôs aeroespaciais, atmosféricos ou submarinos) ou em uma superfície (robôs terrestres). Neste segundo grupo se enquadram os robôs móveis com duas rodas de tração, os quais são alvo de estudos apresentados nesta dissertação. Dentre os diversos sistemas de instrumentação e controle necessários para o correto funcionamento de um robô deste tipo, este trabalho se concentra no controle de seguimento de trajetórias de referência. Serão apresentadas a teoria e os testes de uma estratégia de controle não linear com adaptação de parâmetros em tempo real.

Conforme será apresentado através de exemplos na próxima seção deste trabalho, o controle preciso de seguimento de trajetórias é essencial para o bom desempenho de robôs móveis em ambientes reais, tal como militares, industrial, rural, doméstico, dentre outros.

### 1.1 Exemplos de Robôs Móveis Comerciais de uso Industrial e Militar

Desenvolvidos pela empresa alemã Telerob, os robôs tEODor E Telemax, mostrados na Figura 1-1 (a) e (b) respectivamente, foram adquiridos recentemente, pelo Departamento de Engenharia e Construção (DEC) do Exército Brasileiro e serão destinados à implementação do sistema anti Dispositivo Explosivo Improvisado (anti-DEI), visando atender às demandas operacionais e peculiares em grandes eventos, tal como os Jogos Olímpicos e Paralímpicos Rio 2016 (Portal R3, 2016). São robôs operados remotamente por especialistas em desarmamento de explosivos, portanto devem possuir alto grau de precisão nos comandos executados.

**Figura 1-1: Robôs desenvolvidos pela empresa alemã Telerob. (a) À esquerda, o Telemax com seu braço articulado subindo a escada. (b) À direita, o tEODor com acessório de raio-X.**

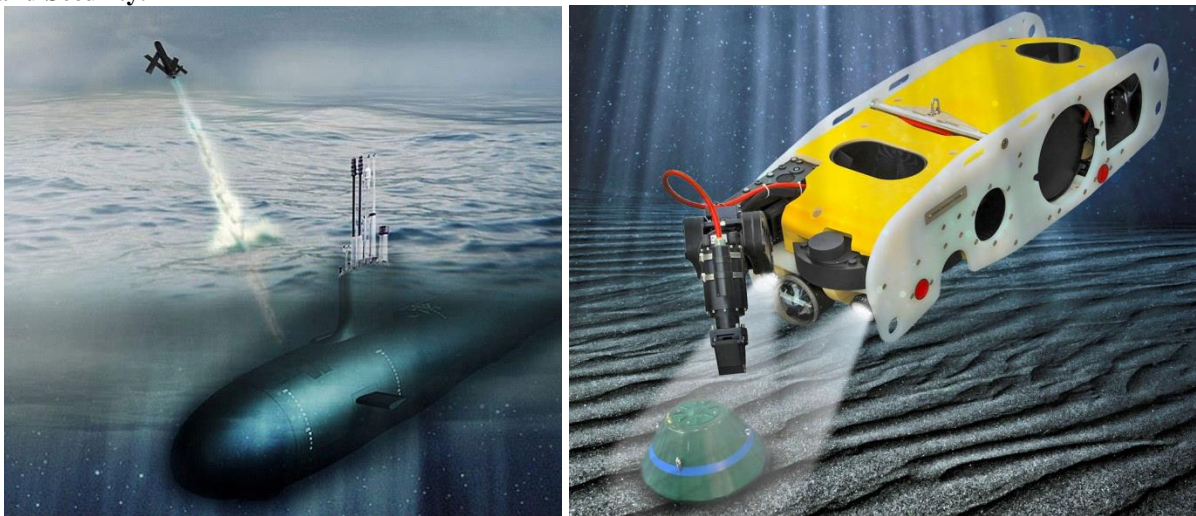


Fonte: Site <http://www.portalr3.com.br>.

Já a Marinha dos Estados Unidos, em resposta aos recentes avanços tecnológicos em mísseis balísticos antinavios da China, anunciou que planeja equipar seus submarinos com Drones Espiões ou UAS (*Unmanned Aircraft System* - Sistema de Aeronave não Tripulada) desenvolvidos pela *AeroVironment Inc.*. Chamados de "*Backwings*" - Figura 1-2 (a) - os robôs medem cerca de 20 polegadas de comprimento, pesam 4 libras, e, segundo a empresa, têm capacidade de voo de 60 minutos (Daily Mail, 2016). Outro robô de interesse da marinha e da polícia americana (FBI, do inglês *Federal Bureau of Investigation*) é o "*Sea Wasp*", Figura 1-2 (b). Projetado pela *Saab Defence and Security*, trata-se de um robô submarino controlado remotamente que utiliza um sonar e um braço manipulador para encontrar, mover, e desarmar explosivos (Engadget, 2016).

Além dos robôs utilizados nas tarefas militares (exploração de ambientes, espionagem, desarmamento de bomba, e etc.), existem aqueles usados em ambientes industriais, comerciais, hospitalares, rurais e domésticos. Robôs manipuladores que realizam operações de soldagem e montagem são bastante disseminados no ambiente industrial, mas também é possível citar exemplos de robôs móveis, como o robô chamado *PIG* (do inglês, *Pipe-line Inspection Gauge*) que são dispositivos que se deslocam dentro dos dutos empurrados pela própria vazão do fluido, realizando funções de limpeza, separação de produtos, remoção de água, desamassamento dos dutos, e fornecimento de informações das condições dos dutos, extensão e localização dos defeitos das tubulações. Atualmente existem mais de 350 tipos de PIGs desenvolvidos para as diferentes necessidades das empresas operadoras de dutos (SOUZA, 2003). Na Figura 1-3, é mostrado um tipo de PIG usado pela PETROBRÁS.

**Figura 1-2:** (a) À esquerda, o robô drone Backwing desenvolvido pela empresa AeroVironment Inc. e adquirido pela Marinha Americana. (b) À direita, o submarino Sea Wasp projetado pela Saab Defence and Security.



Fontes: (a) Site <http://www.dailymail.co.uk> (b) Site <http://www.engadget.com>

**Figura 1-3: Robô PIG instrumentado para tubulações de 14" da PETROBRAS.**



**Fonte: DIAZ, 2008.**

Avanços na tecnologia robótica têm ocorrido em detrimento, principalmente, do desenvolvimento de novos recursos de *hardware*, com dispositivos cada vez menores, baratos, robustos e eficientes, refletindo diretamente nos avanços na área de *software*, pois uma maior quantidade e precisão de informações adquiridas sobre o ambiente de atuação do robô permitem maior complexidade de algoritmos nas áreas de controle, fusão sensorial, tomada de decisão, processamento de sinais e imagens, reconhecimento de voz, entre outros (ROMERO, PRESTES, *et al.*, 2014).

Com isso, novos sistemas de controle para robôs móveis vêm sendo desenvolvidos a fim de melhorar o desempenho e a estabilidade das malhas de controle. Para tal, trabalhos científicos propõem deferentes modelos, topologias e técnicas de controle. Portanto, a fim de avaliar o enquadramento deste trabalho no meio científico, na próxima seção será apresentada uma pesquisa bibliográfica sobre a aplicação de técnicas de controle adaptativo no problema de seguimento de trajetórias por robôs móveis.

## **1.2 Levantamento do Estado da Arte**

No âmbito de sistemas de controle para robôs móveis, a navegação e orientação são de primordial importância no desenvolvimento de sistemas inteligentes autônomos, atraindo o interesse de diversas linhas de pesquisas que buscam resolver problemas que abrangem esse tema (Yang & Kim, 1999). O problema de planejamento e seguimento de trajetórias por robôs móveis é um dos ramos que vem sendo abordado em trabalhos científicos nos últimos anos.

Os métodos para controle de trajetória podem ser divididos em três categorias, sendo classificados como um problema de controle de rastreamento de nível cinemático, de nível dinâmico e a combinação desses dois (Normey-Rico, Gómez-Ortega, & Camacho, 1999).

A finalidade de um controlador baseado na cinemática é enviar sinais de controle de velocidade visando fazer o erro de trajetória convergir para zero. O controlador *backstepping* é um dos mais antigos controladores de nível cinemático (Kanayama, 1990). Em (Júnior & Hermely, 2003), é utilizado um modelo cinemático com ponto de guiamento não coincidindo com ponto médio do eixo das rodas traseiras do robô móvel para provar a estabilidade local.

Quanto à estratégia de controle com base na dinâmica, controla-se a velocidade do sistema de tal modo que tenda a convergir para a velocidade desejada. Conforme a dinâmica do sistema é tratada, essa estratégia pode ser dividida em quatro grupos de controladores: controladores adaptativos, controladores robustos, controladores adaptativos robustos e controladores com base na linearização do modelo.

Os controladores adaptativos dinâmicos são uma extensão adaptativa do controlador *backstepping* cinemático para lidar com parâmetros desconhecidos. Um estudo dessa classe de controlador dinâmico é encontrado em (Huang, 2009). Já os controladores robustos são projetados para lidar com incertezas e perturbações, conforme aplicado em (P. Coelho, 2005). Quanto aos controladores adaptativos robustos com base na dinâmica, são usadas técnicas de inteligência artificial. Em (T. Das, 2006), as incertezas à cerca das não linearidades do sistema e da variação dos parâmetros do sistema são estimadas por uma técnica de lógica *fuzzy*, onde o controlador atualiza a cada interação os parâmetros estimados do sistema não linear. O último grupo de controladores dinâmicos é o dos controladores com base na linearização da dinâmica, utilizados para compensar a dinâmica não linear do sistema. Em (Kühne, 2004), o modelo é linearizado levando em consideração o erro do robô de referência e o robô real. Em (Vieira, 2005), são apresentadas estratégias de controle de posição e orientação de um RMA não-holonômico se baseando apenas em informações de configuração do robô coletadas a partir de um sistema de posicionamento absoluto (posição e orientação no espaço cartesiano  $x$ ,  $y$  e  $\theta$ ), representando o modelo dinâmico na forma linear.

Quanto à estratégia de controle que combina o rastreamento de nível cinemático com o de nível dinâmico, é projetado um controlador para estabilizar a trajetória, fazendo o erro de trajetória convergir para zero, e, em seguida, projeta-se um controlador de velocidade com base na dinâmica dos atuadores, para que a velocidade do robô móvel tenda a convergir



para a velocidade desejada. Em (Fierro & Lewis, 1995), a estabilidade assintótica de um robô móvel usando *backstepping* é provada pelo Método Indireto de Lyapunov a fim de tornar possível a interação de um controlador cinemático e um controlador de torque. Em (Kuc, Baek, & Park, 2001), é proposto um controlador adaptativo para a dinâmica de velocidade dos motores e um outro controlador com base na cinemática estabilizar a direção do robô móvel. Em (Andrade, 2011), é utilizada uma configuração em cascata constituída com uma malha externa com um controlador cinemático monovariável desacoplado para o controle de posição que fornece a velocidade de referência dos motores, e uma malha interna com um controlador preditivo dinâmico multivariável para o controle de velocidade dos motores.

Entre as estratégias de controle preditivo, existem diversas técnicas. Em (Hedjar, 2009), utiliza-se o controlador MPC para determinar o esforço do atuador a partir da estimação do erro com predição em uma amostra. Outro método de controle, GPD (*Generalized Predictive Control*), é estudado em (ANDRADE e TORRICO, 2011), onde foram comparados os desempenhos de um controlador PID e um GPC aplicados ao controle de trajetória de um pequeno RMA projetado pelos próprios autores. Em (Ogawa, 2014), um controlador preditivo baseado em modelo (*Model Predictive Control* - MPC) linearizado é aplicado para o seguimento de trajetória de um RMA, e tem seu desempenho comparado a dois controladores não lineares: um controlador de ganhos fixos proposto em (KLANCAR, 2005); e um controlador preditivo EPSAC (*Extended Prediction Self Adaptive Control*, traduzindo fica Controle Preditivo Autoajustável Estendido). A autora deduz que os controladores preditivos podem ser aplicados de forma eficiente em plantas com modelos multivariáveis em situações na qual a trajetória futura de referência é conhecida e em sistemas com restrições nas variáveis de entrada e de saída. Porém, a principal desvantagem do MPC linearizado é o alto custo computacional o que limita as aplicações práticas.

A fim de buscar alternativas com custo computacional não tão elevado quanto ao dos controladores preditivos e ainda garantir a convergência assintótica do robô para diferentes trajetórias de referência, várias outras propostas de controladores adaptativos têm sido desenvolvidas. Em (S. Dong, 2002), é proposto um controlador adaptativo sincronizado que garante a convergência assintótica para zero do erro de posição entre as rodas do robô e, em seguida, regula o ângulo de orientação do robô para que este consiga seguir a trajetória desejada. Em (CHEN 2011), é proposto um modelo de controlador adaptativo *backstepping* que combina, explicitamente, as restrições cinemáticas e a dinâmica móvel de um robô móvel para rastreamento de trajetórias. Já em (Amorim, 2012), é desenvolvido um controlador adaptativo *backstepping* a estrutura variável (*Variable Structure Adaptive Backstepping*

*Controller*, VS-ABC), onde um Observador Adaptativo de Luenberger substitui os filtros K para plantas monovariáveis, lineares e invariantes no tempo com grau relativo unitário.

Outro problema encontrado no controle de trajetória de um RMA é a inexistência de realimentação de estado puro para a estabilização assintótica de configurações fixas invariantes no tempo, segundo o Teorema de Brockett (2007). Para contornar tal problema, normalmente, é utilizado estimadores *online* que permitem a atualização paramétrica de forma constante. Em (Bertol, 2015), são utilizados métodos de controle com base em lógica *fuzzy* com modelo Takagi-Sugeno (TS-Fuzzy) para substituir incertezas na dinâmica nos modelos de RMA.

Com a mesma finalidade de superar incertezas na dinâmica e parâmetros de um RMA, é proposto em (Mohareri, Dhaouadi, & Rad, 2012) um controlador *backstepping* adaptativo indireto para rastreamento de trajetórias, no qual a estratégia de adaptação é baseada no clássico algoritmo do gradiente descendente. Para a atualização dos ganhos do controlador em tempo real, o cálculo da matriz jacobiana do método do gradiente descendente foi realizado de duas formas. A primeira estratégia foi o cálculo analítico utilizando os modelos cinemático e dinâmico do robô, na qual é necessário o conhecimento de parâmetros físicos do robô. Como os valores destes parâmetros possuem incertezas, a segunda proposta para o cálculo da jacobiana consiste de um estimador baseado em rede neurais. O trabalho compara o desempenho do *backstepping* adaptativo baseado em redes neurais com o desempenho do controlador a parâmetros fixos. Embora seja demonstrado no artigo a metodologia matemática, são omitidos os resultados com a estratégia de adaptação utilizando os cálculos analíticos da jacobiana.

Diante da omissão do trabalho de (Mohareri, Dhaouadi, & Rad, 2012), uma das principais contribuições desta dissertação é realizar a implementação e testes da estratégia de adaptação analítica para o sistema de controle de seguimentos de trajetórias em um robô não-holonômico móvel real. Serão avaliadas diferentes trajetórias de referência e distúrbios.

### **1.3 Objetivos do Trabalho**

Os objetivos que guiaram o desenvolvimento desde trabalho estão divididos em principal e específicos, descritos a seguir.

### 1.3.1 *Objetivo Principal*

Projeto e implementação de um controlador adaptativo não linear para o problema de controle de seguimento de trajetória por um robô móvel não-holonômico com acionamento diferencial de duas rodas.

### 1.3.2 *Objetivos Específicos*

Para atingir o objetivo principal proposto, seguiu-se os seguintes objetivos específicos:

- Modelar o sistema não-holonômico e não linear com base nas equações da dinâmica dos atuadores e do modelo cinemático do robô móvel;
- Estudar e projetar um controlador adaptativo *backstepping* não linear autoajustável para seguimento de trajetórias;
- Estudar e projetar um controlador *backstepping* clássico (com parâmetros fixos) não linear para seguimento de trajetórias;
- Estudar e projetar um controlador de ganhos fixos proposto em (KLANCAR, 2005) para seguimento de trajetórias;
- Analisar os controladores através de resultados de simulação e comparar para diferentes trajetórias;
- Implementar experimentalmente os controladores propostos e comparar os resultados para diferentes trajetórias.

## 1.4 Estrutura da Dissertação

O trabalho está organizado da seguinte forma. No Capítulo 2, são apresentados conceitos e a modelagem adotada neste trabalho para os de robôs móveis não-holonômicos. As técnicas de controle não-linear adaptativo são apresentadas no Capítulo 3. A simulação dos resultados e uma comparação do controlador *backstepping* de ganhos fixos com o controlador *backstepping* adaptativo são analisadas no Capítulo 4. E a implementação experimental do algoritmo de controle proposto em um robô projetado é discutida no Capítulo 5. Por fim, no Capítulo 6, são dissertadas as conclusões do estudo realizado e as sugestões para trabalhos futuros relacionados ao tema.

## 2 MODELAGEM DE ROBÔS MÓVEIS NÃO-HOLONÔMICOS

### 2.1 Introdução

Um robô móvel é um dispositivo eletromecânico controlado por um sistema computacional, que através de sensores e atuadores interagem com o ambiente. Esse sistema computacional é composto por uma arquitetura de *hardware* e *software*, a qual possui níveis hierárquicos conforme apresentado na Figura 2-1 (VIEIRA, 2005).

**Figura 2-1: Níveis hierárquicos da arquitetura de navegação de robôs móveis.**



**Fonte: Vieira, 2005.**

Tais níveis possuem as seguintes funções:

- Nível de Percepção: composto por algoritmos para tratamento das informações coletadas pelos sensores, sendo tratadas de forma a ter algum significado para o sistema, sendo, normalmente, a localização do robô no ambiente, a localização de obstáculos e informações sobre a velocidade do robô.
- Nível de Decisão: responsável pela tomada de decisões a partir das informações obtidas através do nível de percepção. Dependendo do tipo de tarefa que se deseja realizar, são utilizadas árvores de decisão ou técnicas de inteligência artificial.
- Nível de Planejamento de Trajetória: é planejado o melhor caminho que leva o robô de sua localização original a uma determinada localização desejada, levando em consideração os obstáculos e as dimensões do robô. Define-se como localização ou configuração do robô a soma das informações de posição e orientação.

- Nível de Geração de Trajetória: determina as restrições temporais para a execução de um determinado caminho ou trajetória, sendo, portanto, calculadas as velocidades e acelerações que o robô deve adotar em cada instante de tempo para que a execução do caminho seja realizada no período especificado.
- Nível de Sistemas de Controle: responsável por garantir a execução das trajetórias através da geração de sinais de comando para os atuadores. Desta forma, este nível garante que as velocidades e acelerações impostas pelo nível de geração de trajetória sejam executadas de forma correta. Normalmente, os sistemas de controle para robôs móveis são baseados em realimentação de estado, isto é, informações a respeito da velocidade e localização do robô geradas pelo nível de percepção.

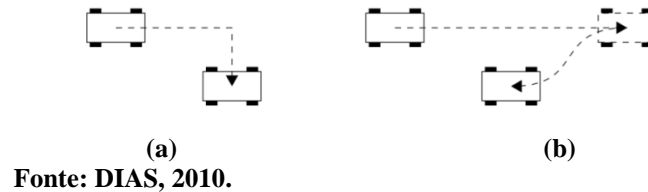
Diante da descrição dos níveis hierárquicos citados, é notável a complexidade envolvida nos projetos e nas implementações de robôs móveis autônomos. As características de movimentação de um robô móvel dependem da disposição e funcionalidade de seus dispositivos de locomoção, que podem restringir seus movimentos ou a realização de trajetórias mais complexas. Tais restrições de movimento são denominadas restrições não-holonômicas (VIEIRA, 2005).

## 2.2 Sistemas não-holonômicos

Partindo da etimologia da palavra holonomia, "*holo*" significa o todo, conjunto, totalidade, e "*nomia*" significa lei. Portanto, o termo holonômico significa “universal”, “integral”, “integrável”. Assim, sistemas não-holonômicos podem ser interpretados como sistemas não-integráveis. Definem-se como não-holonômicos sistemas com dimensão finita onde algum tipo de restrição é imposta a um ou mais estados do sistema (FIGUEIREDO e JOTA, 2004).

No caso de robôs móveis com acionamento diferencial, as restrições são impostas pela impossibilidade de seus atuadores realizarem tais movimentos, impedindo o robô de se movimentar em todas as direções (DIAS, 2010). A ilustra dois tipos de robôs móveis, um holonômico e outro não-holonômico, executando a ação de estacionamento, a fim de evidenciar a restrição não-holonômica que impede um dos robôs de realizar movimentos normais ao eixo das rodas.

**Figura 2-2: Estacionamento de um robô móvel (a) holonômico e (b) não-holonômico.**



### 2.3 Modelagem Matemática de um Robô Móvel Não-holonômico

A modelagem de uma unidade diferencial de um robô móvel consiste em três etapas, a modelagem cinemática, a modelagem dinâmica e a modelagem dos atuadores. Na modelagem cinemática utiliza-se as relações geométricas (álgebra vetorial e matricial) que regem o sistema e os estudos matemáticos do movimento (sem considerar as forças externas) para representar e descrever a localização de um objeto no espaço a partir das equações básicas da cinemática em relação a um sistema de referência (KLANCAR et al., 2005). Por outro lado, na modelagem dinâmica, é feito o estudo matemático do movimento considerando forças e energias externas, as quais são modeladas e incluídas nas equações básicas da dinâmica do sistema (PIERI, 2002). A modelagem do atuador é necessária para achar a relação entre o sinal de saída do controlador e o respectivo sistema mecânico de atuação.

O robô móvel considerado neste trabalho foi projetado e montado por alunos da graduação e pós-graduação em Engenharia Elétrica da Universidade Federal do Ceará - Campus do Pici. Trata-se de um robô móvel não-holonômico sob quatro rodas, sendo apenas rodas tracionadas e de ação diferencial, comandadas por um *driver* embarcado dedicado e com um controlador PID atuando em malha fechada com os motores das duas rodas.

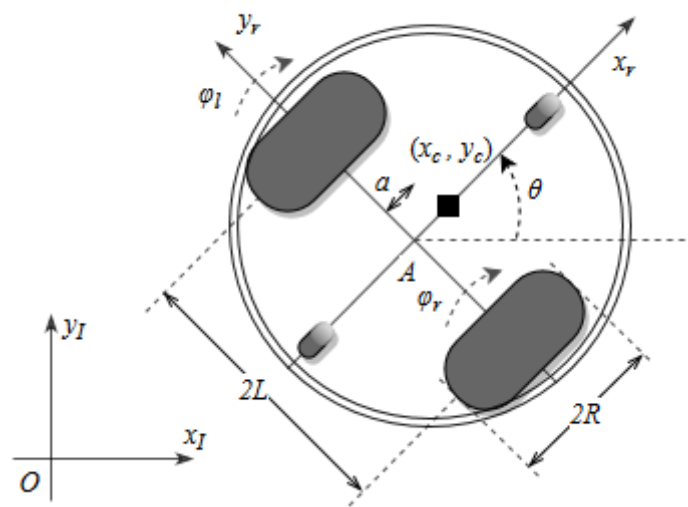
É mostrada na Figura 2-3 uma unidade diferencial do robô móvel utilizado no trabalho. O qual possui duas rodas com raio  $R$ , dispostas a uma distância  $L$  do centro do robô. O centro de massa da plataforma  $(x_c, y_c)$  foi projetado para estar localizado a uma distância  $a$  do centro do eixo das rodas motrizes, denominado ponto A. A ordenada  $(A, x_r, y_r)$  é o ponto de referência fixo para o robô móvel e a ordenada  $(O, x_I, y_I)$  é o ponto de referência inercial global.

A plataforma do robô tem uma massa total  $m$  e um momento de inércia  $I_c$  em torno do seu centro de massa. E a configuração pode ser descrita por cinco coordenadas generalizadas ou também chamadas de variáveis de estado do sistema:

$$q(t) = [x_c(t) \quad y_c(t) \quad \theta(t) \quad \varphi_r(t) \quad \varphi_l(t)]^T, \quad (1)$$

onde  $(x_c, y_c)$  são as coordenadas do centro de massa no ponto de referência inercial,  $\theta$  é o ângulo de posição do robô, e  $(\varphi_r, \varphi_l)$  são os ângulos das rodas motrizes direita e esquerda. Tais coordenadas são variáveis no tempo, contudo, para fins de simplificação, a notação de dependência do tempo não será explícita nestas variáveis.

**Figura 2-3: Modelo dos atuadores diferenciais do robô móvel e pontos de referência.**



Fonte: Autor.

A velocidade de translação  $u$  e a velocidade angular  $\omega$  da plataforma no ponto local podem ser calculadas em função das velocidades dos lados direito e esquerdo das rodas motrizes:

$$u = \frac{R}{2}(\dot{\varphi}_r + \dot{\varphi}_l), \quad (2)$$

$$\omega = \dot{\theta} = \frac{R}{2L}(\dot{\varphi}_r - \dot{\varphi}_l). \quad (3)$$

Para representar as condições práticas de não deslocamento lateral, tal restrição não-holonômica da unidade diferencial pode ser equacionada da seguinte forma:

$$\dot{y}_c \cos \theta - \dot{x}_c \sin \theta - a\dot{\theta} = 0, \quad (4)$$

onde  $\dot{y}_c$  e  $\dot{x}_c$  são os componentes de velocidade do centro de massa no ponto de referência inercial  $A$ , o que significa que a velocidade desse ponto é nula ao longo do eixo ortogonal local e será sempre na direção do eixo de simetria.

Outra restrição não-holonômica considerada é que não há derrapagem das rodas do robô, ou seja, tem-se um rolamento puro das rodas, sendo definido para a roda direita e esquerda respectivamente:

$$\dot{x}_c \cos \theta + \dot{y}_c \sin \theta + L\dot{\theta} = R\dot{\phi}_r, \quad (5)$$

$$\dot{x}_c \cos \theta + \dot{y}_c \sin \theta - L\dot{\theta} = R\dot{\phi}_l. \quad (6)$$

Essas três restrições não-holonômicas podem ser reunidas na seguinte forma matricial:

$$A(q)\dot{q} = 0, \quad (7)$$

onde:

$$A(q) = \begin{bmatrix} -\sin \theta & \cos \theta & -a & 0 & 0 \\ \cos \theta & \sin \theta & L & -R & 0 \\ \cos \theta & \sin \theta & -L & 0 & -R \end{bmatrix} \quad (8)$$

$$\dot{q} = [\dot{x}_c \quad \dot{y}_c \quad \dot{\theta} \quad \dot{\phi}_r \quad \dot{\phi}_l]^T. \quad (9)$$

A energia baseada na abordagem de *Lagrange* pode ser utilizada para encontrar o modelo dinâmico do robô móvel, sendo representado sob a forma geral (YANG e KIM, 1999):

$$M(q)\ddot{q} + V_m(q, \dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d = B(q)\tau - A^T(q)\lambda, \quad (10)$$

onde  $M(q)$  é a matriz de inércia simétrica positiva,  $V_m(q, \dot{q})$  é a matriz das pseudoforças centrípeta e de Coriolis,  $F(\dot{q})$  é a matriz de atrito superficial do robô,  $G(q)$  é o vetor gravitacional,  $\tau_d$  é denotado distúrbios limitados desconhecidos incluindo dinâmicas não modeladas e não estruturadas,  $B(q)$  é a matriz de transformação de entrada,  $\tau$  é o vetor de entrada,  $A^T(q)$  é a matriz associada às restrições e  $\lambda$  é o vetor de restrição das forças.

A locomoção plana do robô implica em eliminar os termos gravitacionais da equação dinâmica:

$$G(q) = 0. \quad (11)$$



Além disso, assumindo que o atrito superficial do robô é desprezível no sistema, pode-se ter também:

$$F(\dot{q}) = 0. \quad (12)$$

Para fins de controle e simulação, o sistema acima pode ser transformado em uma representação mais adequada, onde o objetivo é eliminar o termo de restrição da equação.

A equação cinemática direta é definida pela seguinte transformação:

$$\dot{q} = \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta} \\ \dot{\phi}_r \\ \dot{\phi}_l \end{bmatrix} = S(q)v(t) = \begin{bmatrix} \cos \theta & -a \sin \theta \\ \sin \theta & a \cos \theta \\ 0 & 1 \\ \frac{1}{R} & \frac{L}{R} \\ \frac{1}{R} & -\frac{L}{R} \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix}, \quad (13)$$

onde  $S(q)$  é a matriz cinemática direta. Uma vez que  $S(q)$  é uma matriz de posto completo, formada por um conjunto de vetores suaves e linearmente independentes, pode-se provar a seguinte relação (FIERRO e LEWIS, 1995):

$$S^T(q)A^T(q) = 0 \quad (14)$$

A Eq. (14) é útil para eliminar o termo de restrição a partir da equação dinâmica principal como mostrado no procedimento seguinte. Derivando a Eq. (13), temos:

$$\ddot{q} = \dot{S}(q)v(t) + S(q)\dot{v}(t). \quad (15)$$

Substituindo a Eq. (15) em (10), resulta na seguinte equação:

$$M(q)[\dot{S}(q)v(t) + S(q)\dot{v}(t)] + V_m(q, \dot{q})S(q)v(t) + \tau_d = B(q)\tau - A^T(q)\lambda \quad (16)$$

$$M(q)\dot{S}(q)v(t) + M(q)S(q)\dot{v}(t) + V_m(q, \dot{q})S(q)v(t) + \tau_d = B(q)\tau - A^T(q)\lambda \quad (17)$$

O passo seguinte para eliminar a matriz de restrição  $A^T(q)\lambda$  é multiplicar a Eq. (17) por  $S^T(q)$ , como segue:

$$\begin{aligned} & [S^T(q)M(q)\dot{S}(q) + S^T(q)V_m(q, \dot{q})S(q)]v(t) + [S^T(q)M(q)S(q)]\dot{v}(t) + S^T(q)\tau_d \\ & = S^T(q)B(q)\tau - S^T(q)A^T(q)\lambda \end{aligned} \quad (18)$$

Como pode ser visto a partir da equação acima, o termo  $S^T(q)A^T(q)$  que multiplica o coeficiente desconhecido de *Lagrange*  $\lambda$  será cancelado, resultando em:

$$\begin{aligned} & [S^T(q)M(q)\dot{S}(q) + S^T(q)V_m(q, \dot{q})S(q)]v(t) + [S^T(q)M(q)S(q)]\dot{v}(t) + S^T(q)\tau_d \\ & = S^T(q)B(q)\tau \end{aligned} \quad (19)$$

Pelas seguintes definições apropriadas, pode-se reescrever a Eq. (19) como se segue:

$$\bar{M}(q)\dot{v}(t) + \bar{V}_m(q, \dot{q})v(t) + \bar{\tau}_d = \bar{B}(q)\tau, \quad (20)$$

onde:

$$\bar{M}(q) = S^T(q)M(q)S(q), \quad (21)$$

$$\bar{V}_m(q, \dot{q}) = S^T(q)M(q)\dot{S}(q) + S^T(q)V_m(q, \dot{q})S(q) \quad (22)$$

$$\bar{\tau}_d = S^T(q)\tau_d \quad (23)$$

$$\bar{B}(q) = S^T(q)B(q) \quad (24)$$

As equações acima representam as equações dinâmicas do robô considerando as restrições não-holonômicas e elas podem ser facilmente transformadas para a seguinte equação matricial simplificada:

$$\begin{bmatrix} m & 0 \\ 0 & ma^2 + I_c \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & -ma\dot{\theta} \\ ma\dot{\theta} & 0 \end{bmatrix} \begin{bmatrix} u \\ \dot{\theta} \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & 1 \\ L & -L \end{bmatrix} \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} \quad (25)$$

Nota-se que o termo  $\bar{\tau}_d$ , que define os distúrbios limitados desconhecidos incluindo dinâmicas não modeladas e não estruturadas, não está incluído na Eq. (25) uma vez que esta representa a dinâmica modelada do sistema, contendo, portanto, apenas termos conhecidos.

A representação da dinâmica do robô será utilizada na estratégia de controle de forma que o controlador baseado na dinâmica irá atuar regulando a velocidade dos motores a partir da Eq. (25), conforme mostrado no próximo capítulo na Figura 3-3, a fim de que esta convirja para a velocidade de referência especificada. Com as equações modeladas com base na cinemática e na dinâmica demonstradas nesse capítulo, será mostrado, no capítulo seguinte, a estratégia de controle adotada para tais.

### 3 DESCRIÇÃO DO CONTROLADOR BACKSTEPPING ADAPTATIVO

#### 3.1 Introdução

Diversos tipos de controladores para promover o seguimento de trajetórias por robôs móveis foram desenvolvidos nas últimas décadas (P. Coelho, 2005). Grande parte dos métodos propostos para esse objetivo estão baseados no cálculo do erro entre a posição atual do robô e o caminho de referência a ser seguido. Tais estratégias estão divididas em três paradigmas de acordo com o modelo que descreve o robô: modelo cinemático (KANAYAMA), por um modelo dinâmico; ou por um modelo cinemático e dinâmico (NORMEY-RICO, GÓMEZ-ORTEGA e CAMACHO, 1999).

A finalidade do controlador baseado na cinemática é fazer com que o erro entre a trajetória de referência e a postura do robô convirja para zero. Em seguida, um controlador baseado na dinâmica do sistema é projetado para controlar a velocidade do robô de tal modo que convirja para a velocidade desejada.

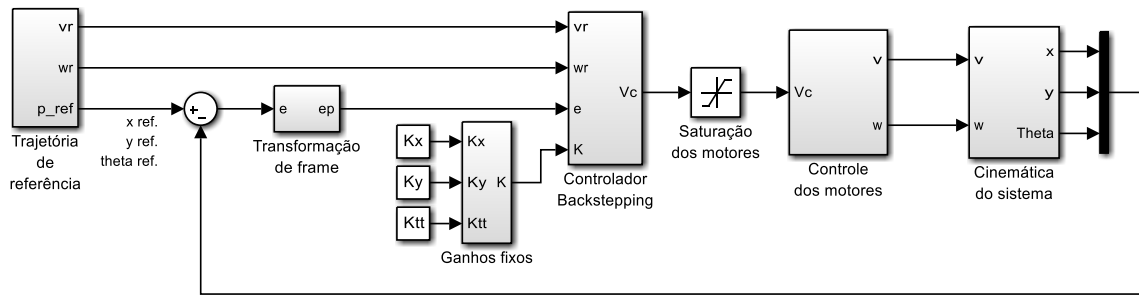
Portanto, o objetivo desse trabalho foi investigar estratégias de controle capazes de lidar com o problema do controle de rastreamento utilizando as equações baseadas na cinemática e na dinâmica do sistema.

#### 3.2 Controlador Não-linear para Rastreamento de Trajetórias

O controlador backstepping proposto neste trabalho é baseado no trabalho de (Mohareri, Dhaouadi, & Rad, 2012). Tal estrutura de controle é uma forma estável de controle de rastreamento para um robô móvel não-holonômico. A estrutura de controle é ilustrada na Figura 3-1.

O modelo de simulação é composto pelos principais blocos: geração de trajetória; transformação de *frame*; controlador backstepping; dinâmica do robô; e cinemática inversa.

**Figura 3-1: Diagrama de blocos do controlador backstepping com ganhos constantes.**



Fonte: Autor.

### 3.2.1 Geração de trajetória

A geração da trajetória de referência é definida através do vetor  $q_r = [x_r \ y_r \ \theta_r]$ , calculado a partir de especificações de velocidade linear ( $v_r$ ) e angular ( $\omega_r$ ) desejadas para cada trajetória.

$$\begin{cases} x_r = \int v_r \cos \theta_r dt \\ y_r = \int v_r \sin \theta_r dt \\ \theta_r = \int \omega_r dt \end{cases} \quad (26)$$

### 3.2.2 Transformação de Frame

Considerando o referencial do robô móvel representado na Figura 2-3, o erro é assim calculado:

$$e_p = \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix} = T_e e_r, \quad (27)$$

onde  $e_r$  é a diferença entre a trajetória de referência  $q_r = [x_r \ y_r \ \theta_r]$  e a postura do robô estimada pelo algoritmo de odometria  $q = [x \ y \ \theta]$ , e a matriz  $T_e$  é a matriz de rotação.

### 3.2.3 Controlador Backstepping

A lei de controle que calcula o vetor de velocidades de referência  $v_c$  do robô é (KANAYAMA, 1990):

$$v_c = \begin{bmatrix} v_r \cos(e_\theta) + K_x e_x \\ \omega_r + K_y v_r e_y + K_\theta v_r \sin(e_\theta) \end{bmatrix}, \quad (28)$$

onde  $v_c = f(e_p, v_r, K)$  e  $K = (K_x, K_y, K_\theta)$ .

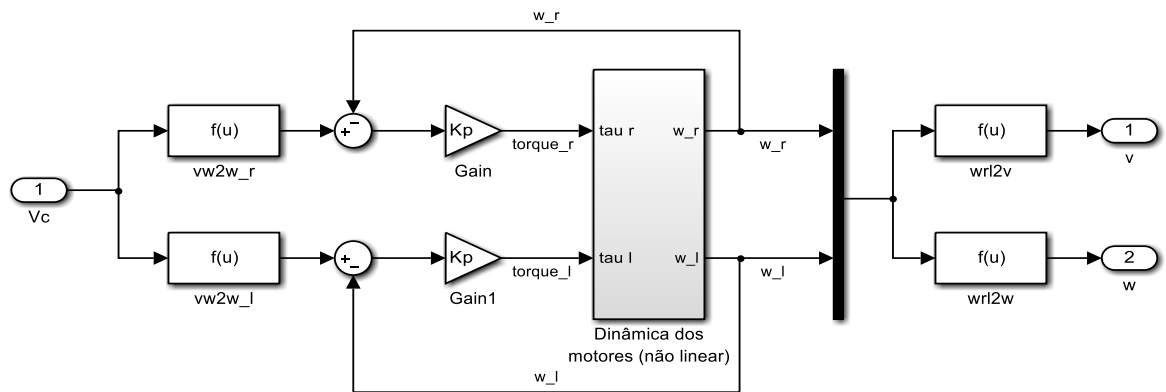
Como pode ser visto nas equações acima, o controlador tem três ganhos os quais são definidos como valores constantes positivos. Os termos  $v_r \cos(e_\theta)$  e  $\omega_r$  são as ações *feedforward* do controlador, enquanto os demais termos compõem a ação *feedback*.

Dentre as características deste controlador não-linear destaca-se a simplicidade de implementação, uma vez que se baseia somente no modelo cinemático. Contudo, é necessário cuidados no ajuste dos ganhos para cada trajetória de referência (KANAYAMA, 1990).

### 3.2.4 Dinâmica do robô

Neste subsistema do modelo de simulação são implementadas as malhas de controle de velocidade das rodas. O controlador PID mostrado na Figura 3-2 é utilizado para fazer a velocidade do robô seguir a velocidade de referência da entrada.

Figura 3-2: Bloco de controle dos motores.



Fonte: Autor.

Em Mohareri (2012) é proposto um controlador PD para o controle de velocidade das duas rodas na forma:

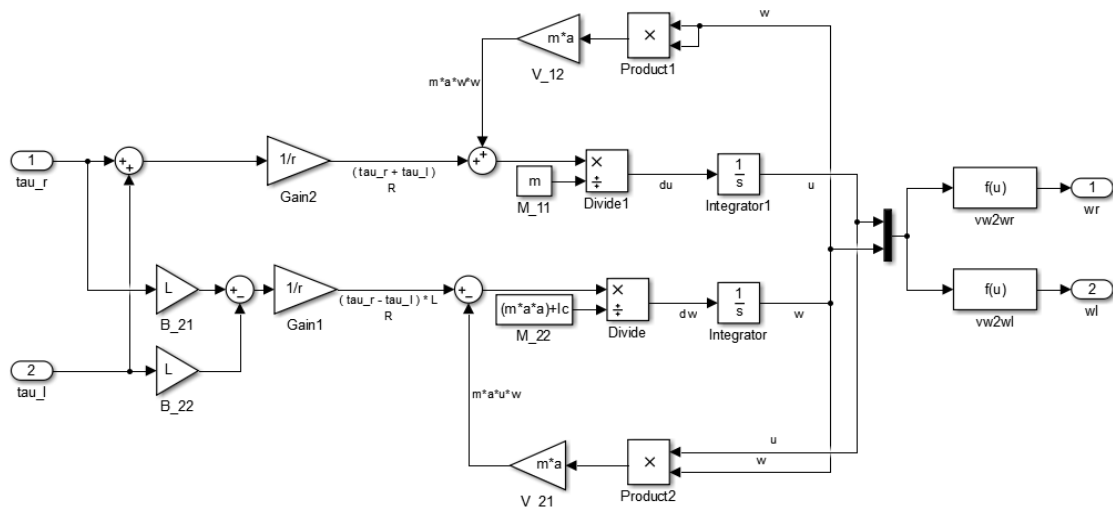
$$\tau = K_p e_c + K_D \frac{de_c}{dt}, \quad (29)$$

onde  $e_c = [e_l \ e_r]$  é um vetor com os erros de velocidade das rodas da esquerda e direita, respectivamente.

$$K_p = \begin{bmatrix} K_{Pr} & 0 \\ 0 & K_{Pl} \end{bmatrix}. \quad (30)$$

O sistema de equações que representa a dinâmica não-linear do robô foi implementado conforme mostrado no diagrama em blocos da Figura 3-3.

Figura 3-3: Diagrama da dinâmica dos motores (não linear).



Fonte: Autor.

### 3.2.5 Cinemática inversa

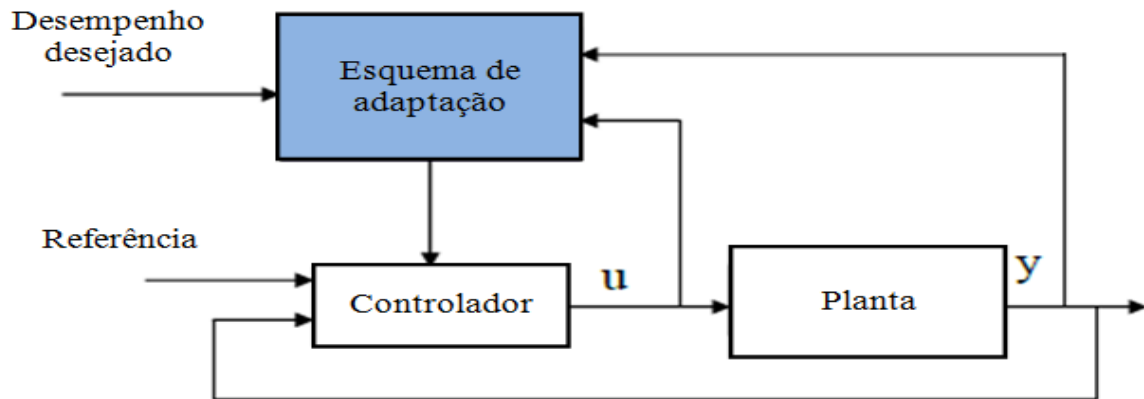
No bloco cinemática do sistema é implementado o algoritmo de odometria, que realiza a estimação do vetor postura do robô  $q = [x \ y \ \theta]$ . De forma semelhante à Eq.(26), os sinais de entrada deste bloco são a velocidade linear ( $v$ ) e angular ( $\omega$ ) do robô, as quais são calculadas a partir da Eq.(2) e Eq.(3), que usam as velocidades angulares medidas através dos *encoders* nas duas rodas ( $\dot{\phi}_d$  e  $\dot{\phi}_e$ ). Estes valores são utilizados como realimentação de postura para o controlador *backstepping*.

$$\begin{cases} x = \int v \cos \theta dt = \int \left( \frac{R}{2} (\dot{\phi}_d + \dot{\phi}_e) \right) \cos \theta dt \\ y = \int v \sin \theta dt = \int \left( \frac{R}{2} (\dot{\phi}_d + \dot{\phi}_e) \right) \sin \theta dt \\ \theta = \int \omega dt = \int \left( \frac{R}{2L} (\dot{\phi}_d - \dot{\phi}_e) \right) dt \end{cases} \quad (31)$$

### 3.3 Estratégia de adaptação online dos ganhos do controlador backstepping

Nesta seção é apresentada a estratégia para a adaptação em tempo real do vetor de ganhos do controlador backstepping. Conforme ilustrado na Figura 3-4, um esquema de adaptação é utilizado a fim de obter um valor de ganhos para o controlador tal que o sistema mantenha a estabilidade e desempenho mesmo com o sistema sujeito a variações paramétricas e variações de condições de operação.

**Figura 3-4: Esquema de controle adaptativo.**



**Fonte:** Adaptado de Aström (Adaptive Control, 2008).

A teoria de controle adaptativo (Astrom, 1999) apresenta diversas estratégias de adaptação, tal como controle adaptativo baseado em modelo (MRAC, *model reference adaptive controller*), auto-ajustável (STR, *self tuning regulator*) e ganhos programados.

A estratégia apresentada neste trabalho é do tipo STR, na qual é utilizada uma função custo que deve ser minimizada. Dentre os algoritmos que utilizam derivadas da função-objetivo para determinar a direção da busca, destacam-se o método de Newton e o método do gradiente descendente (GONÇALVEZ, 2011).

No método de Newton, sucessivas minimizações de aproximações quadráticas são feitas por uma expansão em série de Taylor até a segunda ordem da função-objetivo, convergindo rapidamente. Contudo, o principal problema do método de Newton é a necessidade do cálculo e da inversão da matriz *hessiana*, o que pode ser demorado e sujeito a problemas numéricos (NOCEDAL e WRIGHT, 1999).

Entretanto, no método do gradiente descendente, parte-se de um ponto inicial e se obtém a direção que produz o maior decréscimo da função-objetivo, indicada pelo vetor gradiente (GONÇALVEZ, 2011). Ilustrando o que foi dito, na,  $x^{(k)}$  representa o vetor gradiente de uma função-objetivo.



Assim, para realizar a minimização de uma função, é usado um esquema iterativo, onde em cada passo se toma a direção negativa do vetor gradiente, haja visto que o gradiente corresponde à direção de maior crescimento da função. A partir do tamanho do passo pré-determinado pelo usuário ou obtido por algum método de busca linear, a próxima solução é calculada e o processo se repete até que o algoritmo não obtenha ganhos significativos (GONÇALVEZ, 2011).

Dessa forma, o algoritmo de controle adaptativo proposto melhora o controlador *backstepping* com ganhos constantes, uma vez que os ganhos passam a ser variáveis. A estrutura de controle proposta adapta o ganho do controlador cinemático base para minimizar a função custo:

$$F = \frac{1}{2} \sum \gamma_x e_x^2 + \gamma_y e_y^2 + \gamma_\theta e_\theta^2, \quad (32)$$

onde as constantes  $\gamma_x$ ,  $\gamma_y$  e  $\gamma_\theta$  são ponderações para os respectivos erros de posição  $e_x$ ,  $e_y$  e  $e_\theta$ .

Definindo-se  $P(\alpha) = P(K_x, K_y, K_\theta)$  como uma solução (ótimo) da função  $F$ , o gradiente  $\nabla(F) = 0$  é chamado de ponto estacionário da função. Considerando-se o tempo discreto com índice  $k$  sendo a iteração atual, o gradiente de  $F$  é calculado por:

$$\nabla(F) = \frac{\partial F}{\partial \alpha} = \alpha_{k+1} - \alpha_k \quad (33)$$

onde  $\alpha_{k+1}$  é o vetor solução da próxima iteração e  $\alpha_k$  é solução no instante atual  $k$ . Portanto os valores de  $\alpha = K_x, K_y, K_\theta$  são atualizados de acordo com:

$$\alpha_{k+1} = \alpha_k + \nabla(F) \quad (34)$$

Porém, por definição, o gradiente aponta para o máximo da função  $F$ . Como o objetivo é a minimização dos erros, é necessário a convergência no sentido contrário do gradiente:

$$-\nabla(F) = -\frac{\partial F}{\partial \alpha} = \alpha_{k+1} - \alpha_k \quad (35)$$

Então neste caso os valores de  $\alpha = K_x, K_y, K_\theta$  são atualizados de acordo com:

$$\alpha_{k+1} = \alpha_k - \nabla(F) \quad (36)$$

A fim de se ajustar o passo do algoritmo do gradiente descendente, é utilizado um termo adicional  $\eta$  ponderando o gradiente:

$$\alpha_{k+1} = \alpha_k - \eta \nabla(F) \quad (37)$$

Substituindo-se os devidos ganhos  $\alpha = K_x, K_y, K_\theta$  na função  $F$ , os ganhos do controlador são atualizados pelas seguintes equações ( $n=k$ ):

$$K_x(n) = K_x(n-1) - \eta_x \frac{\partial F}{\partial K_x}, \quad (38)$$

$$K_y(n) = K_y(n-1) - \eta_y \frac{\partial F}{\partial K_y}, \quad (39)$$

$$K_\theta(n) = K_\theta(n-1) - \eta_\theta \frac{\partial F}{\partial K_\theta}, \quad (40)$$

onde  $\eta_x, \eta_y$  e  $\eta_\theta$  são os passos do algoritmo de gradiente descendente.

Os ganhos do controlador de base cinemática são considerados parte da função custo e são otimizados e atualizados de acordo com o método de gradiente descendente. Para o cálculo dos termos  $\partial F/\partial K_x, \partial F/\partial K_y$  e  $\partial F/\partial K_\theta$ , é necessário calcular a derivada parcial da função custo em relação vetor de parâmetros:

$$\frac{\partial F}{\partial \alpha} = \gamma_x e_x \frac{\partial e_x}{\partial \alpha} + \gamma_y e_y \frac{\partial e_y}{\partial \alpha} + \gamma_\theta e_\theta \frac{\partial e_\theta}{\partial \alpha} = e_p^T \left( \Gamma \frac{\partial e_p}{\partial \alpha} \right), \quad (41)$$

onde:

$$\Gamma = \begin{bmatrix} \gamma_x & 0 & 0 \\ 0 & \gamma_y & 0 \\ 0 & 0 & \gamma_\theta \end{bmatrix},$$

Substituindo a Eq. (27) na Eq. (41), temos:

$$e_p = \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = T_e(q_r - q),$$

$$\frac{\partial F}{\partial \alpha} = e_p^T \Gamma \frac{\partial(T_e(q_r - q))}{\partial \alpha} = -e_p^T \Gamma T_e \frac{\partial q}{\partial \alpha}. \quad (42)$$

Usando a regra da cadeia, a derivada  $(\partial q / \partial \alpha)$  pode ser escrita como:

$$\frac{\partial q}{\partial \alpha} = \frac{\partial q}{\partial v_c} \frac{\partial v_c}{\partial \alpha}. \quad (43)$$

A primeira derivada parcial da equação acima,  $(\partial q / \partial v_c)$ , é definida como a matriz Jacobiana com relação às entradas de velocidade do sistema:

$$\frac{\partial q}{\partial v_c} = J = \begin{bmatrix} \frac{\partial x}{\partial v_c} & \frac{\partial q}{\partial \omega_c} \\ \frac{\partial y}{\partial v_c} & \frac{\partial y}{\partial \omega_c} \\ \frac{\partial \theta}{\partial v_c} & \frac{\partial \theta}{\partial \omega_c} \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \\ J_{31} & J_{32} \end{bmatrix}. \quad (44)$$

Enquanto que a segunda derivada parcial da Eq.(43),  $(\partial v_c / \partial \alpha)$ , é calculada como:

$$\frac{\partial v_c}{\partial \alpha} = \begin{bmatrix} \frac{\partial v_c}{\partial K_x} & \frac{\partial v_c}{\partial K_y} & \frac{\partial v_c}{\partial K_\theta} \\ \frac{\partial \omega_c}{\partial K_x} & \frac{\partial \omega_c}{\partial K_y} & \frac{\partial \omega_c}{\partial K_\theta} \end{bmatrix} = \begin{bmatrix} e_x & 0 & 0 \\ 0 & v_r e_y & v_r \sin(e_\theta) \end{bmatrix}. \quad (45)$$

Portanto, a derivada desejada da Eq.(43), pode ser encontrada substituindo as Eqs. (44) e (45) na Eq.(43):

$$\frac{\partial q}{\partial \alpha} = J \begin{bmatrix} e_x & 0 & 0 \\ 0 & v_r e_y & v_r \sin(e_\theta) \end{bmatrix} \quad (46)$$

Sendo assim, a derivada da função custo com relação aos ganhos do controlador Eq. (41) fica:

$$\frac{\partial F}{\partial \alpha} = -e_p^T \Gamma T_e J \begin{bmatrix} e_x & 0 & 0 \\ 0 & v_r e_y & v_r \sin(e_\theta) \end{bmatrix} = \begin{bmatrix} \frac{\partial J}{\partial K_x} & \frac{\partial J}{\partial K_y} & \frac{\partial J}{\partial K_\theta} \end{bmatrix} \quad (47)$$

A principal parte na derivação da Eq. (47) é o procedimento de calcular a matriz Jacobiana do sistema. A matriz Jacobiana pode ser estimada usando as equações exatas do sistema ou pode ser fornecida por meio de técnicas de Redes Neurais (MOHARERI, DHAOUADI e RAD, 2012).

### 3.4 Cálculo da Jacobiana usando as equações exatas do sistema

A matriz Jacobiana  $J = (\partial q / \partial v_c)$  pode ser calculada usando as equações dinâmicas e cinemáticas do robô. Decompondo-a, temos a seguinte forma:

$$J = \frac{\partial q}{\partial v_c} = \frac{\partial q}{\partial v} \frac{\partial v}{\partial \tau} \frac{\partial \tau}{\partial e_c} \frac{\partial e_c}{\partial v_c}. \quad (48)$$

Para calcular a derivada parcial do primeiro termo do produto mostrado na equação acima, usamos a Eq. (13) da cinemática do robô como segue:

$$\dot{q} = \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & -a \sin \theta \\ \sin \theta & a \cos \theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} = Sv. \quad (49)$$

Integrando a equação acima ao longo do tempo, temos:

$$q = \int \dot{q} dt = \int Sv dt. \quad (50)$$

Portanto, a derivada parcial do primeiro termo do produto é:

$$\frac{\partial q}{\partial v} = \int S dt = \int \begin{bmatrix} \cos \theta & -a \sin \theta \\ \sin \theta & a \cos \theta \\ 0 & 1 \end{bmatrix} dt. \quad (51)$$

A derivada parcial do segundo termo do produto da Eq. (48), pode ser encontrada a partir da Eq. (20) da dinâmica do sistema, onde simplificando-a para a forma compacta, temos que:

$$\bar{M}(q)\dot{v}(t) + \bar{V}_m(q, \dot{q})v(t) = \bar{\tau}. \quad (52)$$

onde  $\bar{\tau} = \bar{B}(q)\tau - \bar{\tau}_d$  é o torque equivalente total.

Isolando a derivada do vetor velocidade na Eq.(52), temos:

$$\dot{v}(t) = -\bar{M}(q)^{-1}\bar{V}_m(q, \dot{q})v(t) + \bar{M}(q)^{-1}\bar{\tau}. \quad (53)$$

Portanto, para encontrar o vetor velocidade, integra-se a Eq.(53):

$$v(t) = \int_0^t -\bar{M}(q)^{-1}\bar{V}_m(q, \dot{q})v(t)dt + \int_0^t \bar{M}(q)^{-1}\bar{\tau} dt. \quad (54)$$

A Eq. (54) mostra que apenas o seu segundo termo depende do vetor torque. Como resultado disso, a matriz Jacobiana desejada é:

$$\begin{aligned} \frac{\partial v}{\partial \tau} &= \begin{bmatrix} \frac{\partial u}{\partial \bar{\tau}_1} & \frac{\partial u}{\partial \bar{\tau}_2} \\ \frac{\partial \omega}{\partial \bar{\tau}_1} & \frac{\partial \omega}{\partial \bar{\tau}_2} \end{bmatrix} = \begin{bmatrix} \int_0^t \bar{M}(q)^{-1}_{11} dt & \int_0^t \bar{M}(q)^{-1}_{21} dt \\ \int_0^t \bar{M}(q)^{-1}_{12} dt & \int_0^t \bar{M}(q)^{-1}_{22} dt \end{bmatrix} \\ &= \begin{bmatrix} \int_0^t \frac{1}{m} dt & 0 \\ 0 & \int_0^t \frac{1}{I_c + ma^2} dt \end{bmatrix} = \begin{bmatrix} \frac{1}{m}t & 0 \\ 0 & \frac{1}{I_c + ma^2}t \end{bmatrix}. \end{aligned} \quad (55)$$

A partir da Eq.(29), o torque de entrada gerado pode ser aproximado por:

$$\tau = K_p e_c. \quad (56)$$

A derivada parcial do controlador PD é desconsiderada na equação acima por ser muito pequena ou desprezível. Assim a derivada parcial do terceiro termo do produto da Eq. (48) é dada por:

$$\frac{\partial \tau}{\partial e_c} = K_p = \begin{bmatrix} K_{P_r} & 0 \\ 0 & K_{P_t} \end{bmatrix}. \quad (57)$$

O próximo termo do produto da Eq. (48), a quarta derivada parcial fica na forma:

$$\frac{\partial e_c}{\partial v_c} = \frac{\partial (v_c - v_a)}{\partial v_c} = 1. \quad (58)$$

Portanto, combinando as equações anteriores, Eqs. (51) (55) (57) e (58), a matriz Jacobiana desejada pode ser encontrada da seguinte forma:

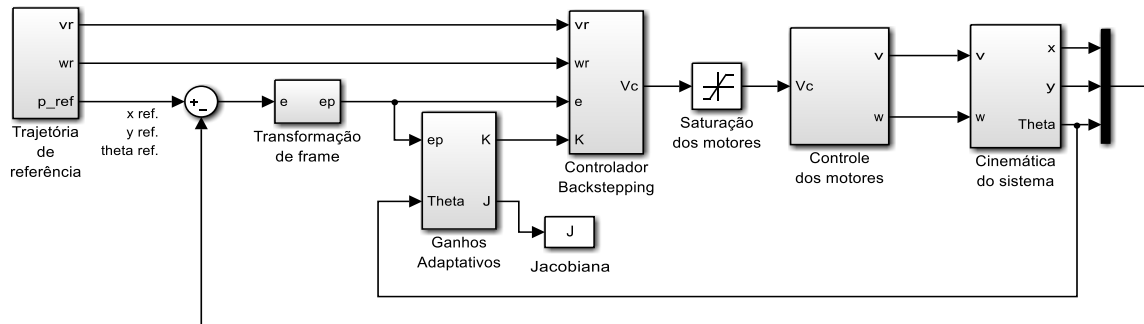
$$J = \int \begin{bmatrix} \cos \theta & -a \sin \theta \\ \sin \theta & a \cos \theta \\ 0 & 1 \end{bmatrix} dt \cdot \int \begin{bmatrix} \frac{1}{m} & 0 \\ 0 & \frac{1}{I_c + ma^2} \end{bmatrix} dt \cdot \begin{bmatrix} K_{P_r} & 0 \\ 0 & K_{P_t} \end{bmatrix} \cdot 1$$

$$J = \int S dt \frac{\partial v}{\partial \tau} \begin{bmatrix} K_{P_r} & 0 \\ 0 & K_{P_t} \end{bmatrix}. \quad (59)$$

Como pode ser visto na equação acima, o cálculo da Jacobiana usando este método requer conhecimento de alguns parâmetros físicos do sistema. Supõe-se também que todos os parâmetros do sistema são certos e fixos, o que não é totalmente verdadeiro, pois os mesmos estão submetidos a parâmetros incertos e distúrbios. Neste trabalho, será avaliada a estratégia de adaptação através de testes de simulação e também em um robô móvel.

Na Figura 3-5 é mostrado um diagrama em blocos do modelo de simulação computacional do robô com o controlador adaptativo.

Figura 3-5: Diagrama de blocos do controlador adaptativo backstepping.



Fonte: Autor.

### 3.5 Discretização para implementação no robô real

A estratégia de adaptação de ganhos proposta na seção anterior está definida no domínio do tempo contínuo. Contudo, para a implementação no robô real foi necessária a discretização dos blocos dinâmicos. Para isto foi utilizado o Método de Euler, no qual a relação entre a variável de Laplace  $s$  e o operador atraso discreto  $q^{-1}$  é dado por:  $s = (1 - q^{-1})/T_s$ , onde  $T_s$  é o intervalo de amostragem igual a 100 ms.

Considerando-se o índice de tempo discreto  $k$ , a matriz  $\partial q/\partial v$  discretizada fica na forma:

$$S(k) = \begin{bmatrix} S_{11}(k) & S_{12}(k) \\ S_{21}(k) & S_{22}(k) \\ S_{31}(k) & S_{32}(k) \end{bmatrix}, \quad (60)$$

onde:

$$S_{11}(k) = S_{11}(k-1) + t_s \cos \theta(k)$$

$$S_{12}(k) = S_{12}(k-1) - t_s a \cdot \sin \theta(k)$$

$$S_{21}(k) = S_{21}(k-1) + t_s \sin \theta(k)$$

$$S_{22}(k) = S_{22}(k-1) + t_s a \cdot \cos \theta(k)$$

$$S_{31}(k) = 0$$

$$S_{32}(k) = kT_s.$$

Outro termo dinâmico que precisa ser discretizado é o cálculo de  $\partial v/\partial \tau$ :

$$\hat{\partial}v/\partial\tau = \begin{bmatrix} (1/m)kT_s & 0 \\ 0 & kT_s / (I_c + ma^2) \end{bmatrix}. \quad (61)$$

A estimativa de postura do robô via odometria utiliza o modelo cinemático do robô móvel. Portanto este bloco dinâmico na implementação real deve ser discretizado na forma:

$$\begin{aligned} \theta(k) &= \theta(k-1) + t_s \omega(k) \\ x(k) &= x(k-1) + t_s v(k) \cos \theta(k) \\ y(k) &= y(k-1) + t_s v(k) \sin \theta(k) \end{aligned} \quad (62)$$

As equações discretizadas foram utilizadas tanto nos modelos de simulação computacional quanto na implementação prática no robô real.

### 3.6 Estratégias de controle não-linear baseada em (Klancar, Matko, & Blazic, 2005)

Para fins de comparação do desempenho do controlador adaptativo apresentado na seção passada, também serão apresentados resultados de uma estratégia de controle não-linear baseada no trabalho de (Kanayama, 1990) e (Klancar, Matko, & Blazic, 2005).

Neste caso o controlador para rastreamento de trajetórias é definido como:

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -k_x & 0 & 0 \\ 0 & -\text{sign}(u_{r1})k_y & -k_\theta \end{bmatrix} \cdot \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix}. \quad (63)$$

onde os ganhos do controlador são calculados de acordo com a trajetória de referência:

$$\begin{aligned} k_x &= k_\theta = 2\xi\omega_n(k) \\ k_y &= g \cdot |u_{r1}(k)|. \end{aligned} \quad (64)$$



E os valores de  $\zeta$  e  $g$  são definidos pelo usuário a fim de ajustar a resposta desejada:

$$\omega_n(k) = \sqrt{u_{r2}^2(k) + gu_{r1}^2(k)} \quad (65)$$

Na próxima seção, serão apresentados resultados de simulações dos controladores propostos para diferentes trajetórias e condições iniciais.

## 4 RESULTADOS DE SIMULAÇÃO

Para a realização das simulações, implementou-se no *software* Matlab uma rotina computacional dos controladores abordados neste trabalho. Com isso, foi possível analisar o desempenho do controlador adaptativo proposto e compará-lo com o desempenho de dois controladores: o controlador *backstepping* clássico (de ganho fixo) e o controlador proposto por Klancar em (KLANCAR, 2005). Foram escolhidas três tipos de trajetórias para a análise, conforme mostrado no decorrer deste capítulo.

Inicialmente, é necessário especificar algumas variáveis e condições iniciais do sistema, sendo os parâmetros físicos adotados para o robô simulado foram:

- massa ( $m$ ):  $20kg$
- raio das rodas ( $r$ ):  $7cm$
- largura do eixo ( $L$ ):  $20cm$
- distância entre o centro do eixo e o centro de massa ( $a$ ):  $1cm$
- momento de inércia ( $I_c$ ):  $m \cdot L^2 = 0,2kg/m^2$

Com os parâmetros do robô definidos, determinou-se que o tempo de amostragem seria de  $100ms$ , o que dá uma boa margem de tempo para o processador atualizar o algoritmo a cada iteração. Em seguida, foi analisado o comportamento do sistema com os valores de ajuste dos controladores apontados em cada artigo em questão, com o intuito de validar o ajuste inicial dos controladores.

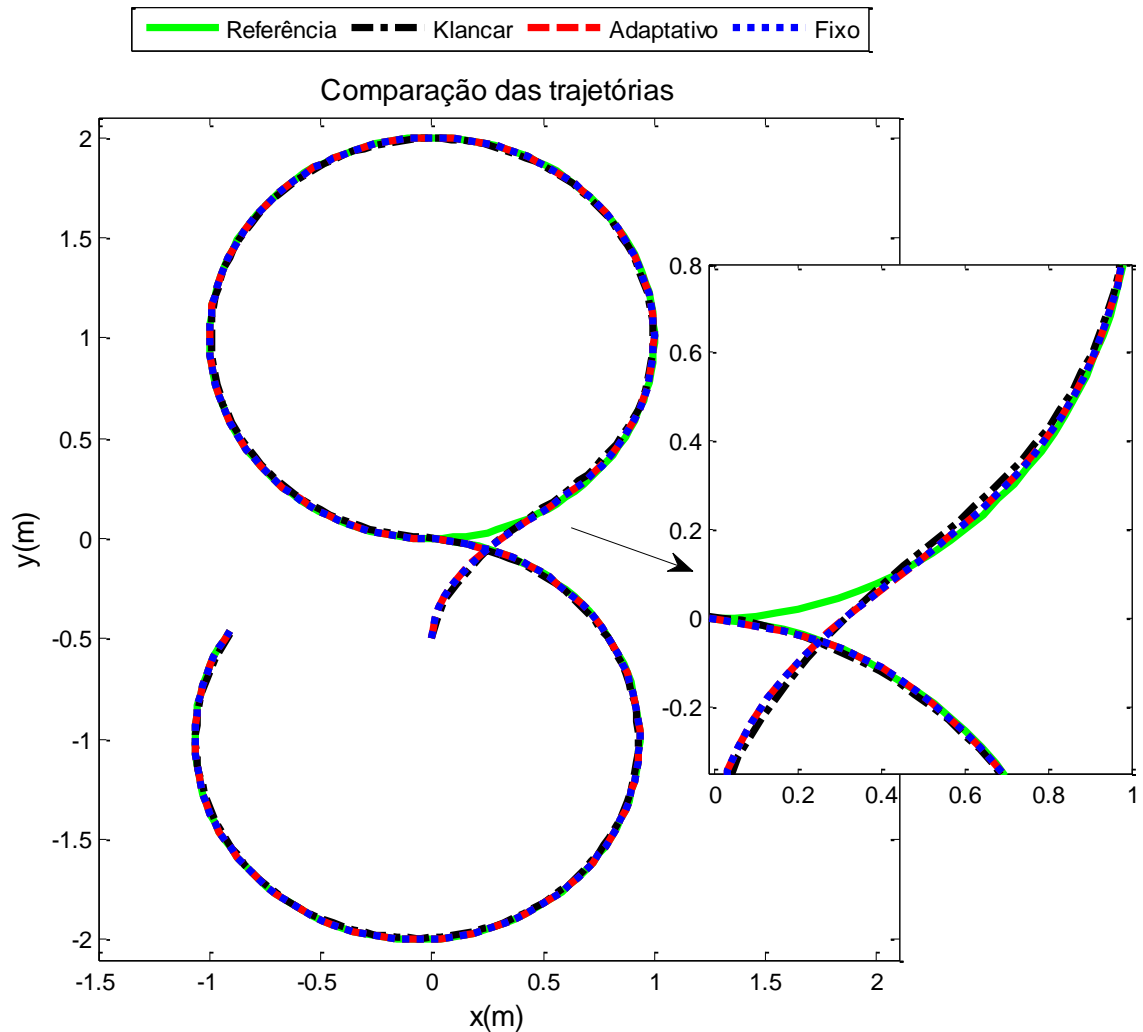
Em (KLANCAR, MATKO e BLAZIC, 2005), os valores do coeficiente de amortecimento  $\zeta$ , da frequência natural  $\omega_n$ , e do coeficiente de grau de liberdade  $g$ , são:  $\xi = 0,6$ ;  $\omega_n = 2$ ;  $g = 30$ , respectivamente. Enquanto que os valores dos ganhos do controlador *backstepping* clássico escolhidos em (MOHARERI, DHAOUADI e RAD, 2012) são:  $K_{x_0} = 1$ ;  $K_{y_0} = 120$ ;  $K_{\theta_0} = 3$ . Os valores escolhidos para os ganhos do controlador *backstepping* clássico foram utilizados como condição inicial dos ganhos do controlador adaptativo.

A trajetória de referência escolhida foi a de um oito para uma a velocidade linear de referência de  $0,5m/s$  e uma velocidade angular de  $0,5rad./s$ . Para formar uma trajetória de um oito completo de raio unitário ( $1m$ ) com as velocidades de referência adotadas, a simulação deveria ser de  $25s$ . Porém, simulou-se até  $23s$ , a fim de que tornar o gráfico mais "limpo", não finalizando a trajetória no mesmo ponto inicial de referência. Considerou-se que o robô estará em uma condição de erro de estado inicial, ou seja, não partindo da origem, e

sim do ponto  $(X_0; Y_0) = (0, 0; -0,5)$ , posicionado a um ângulo  $\theta_0 = \pi/2 \text{ rad.}$  em relação ao eixo de referência (Figura 2-3).

Após a simulação dos controladores para os valores escolhidos nos trabalhos anteriormente referenciados, as trajetórias resultantes estão sobrepostas na Figura 4-1.

**Figura 4-1: Desempenho dos controladores para uma trajetória em oito de raio unitário.**

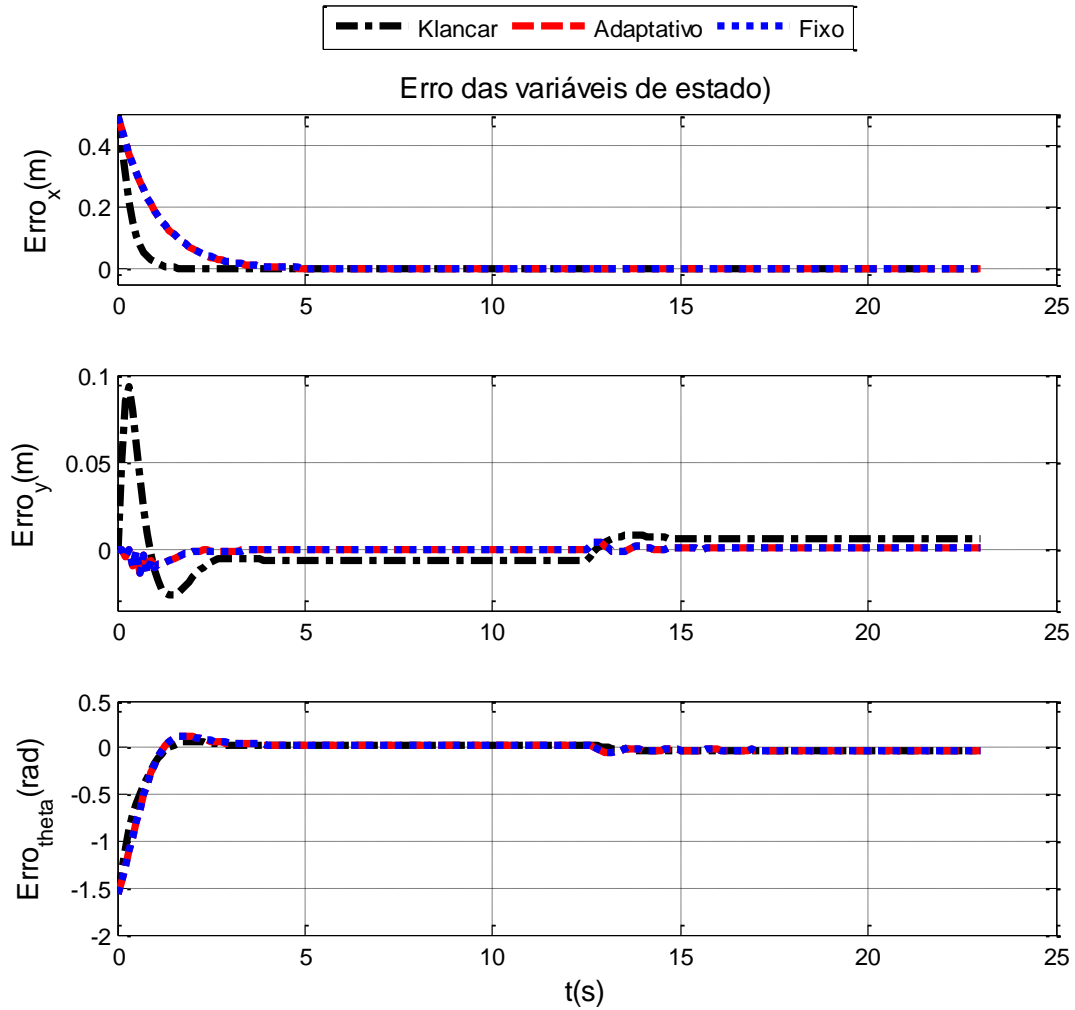


**Fonte: Autor.**

Os desempenhos são, aparentemente, satisfatórios ao analisar a Figura 4-1, ou seja, bem próximos da referência (legenda em verde). Percebe-se que as trajetórias com controlador Klancar (legenda em preto), com controlador *backstepping* (legenda em azul), e com controlador adaptativo (legenda em vermelho) apresentaram desempenhos semelhantes.

Na Figura 4-2 são mostrados os erros das variáveis de estado  $x$ ,  $y$  e  $\theta$ , indicando a diferença entre as trajetórias dos controladores e a trajetória de referência mostrada na Figura 4-1. O período de tempo é o mesmo das trajetórias (23s) e a unidade de medida é o metro.

Figura 4-2: Erro das variáveis de estado  $x$ ,  $y$  e  $\theta$  com controladores ajustados conforme artigos.

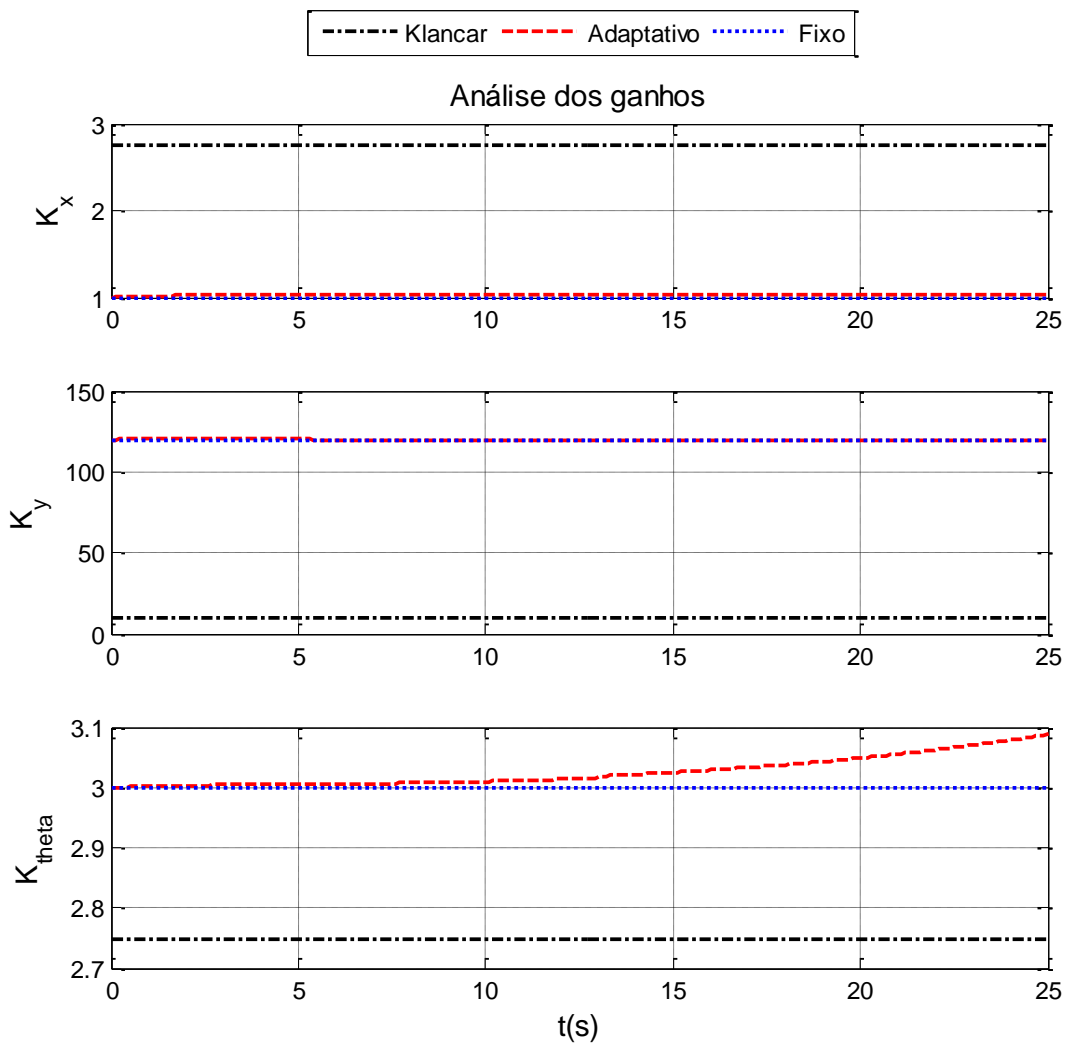


Fonte: Autor.

Ao verificar os erros das variáveis de estado na Figura 4-2, observa-se que ambos convergem dentro de um intervalo de 5s, para valores próximos à zero, na ordem de  $10^{-3}$ , demonstrando um bom desempenho dos controladores no regime permanente.

Já na Figura 4-3, percebe-se que a constância dos valores dos ganhos do controlador *backstepping* clássico, por ser um controlador de ganhos fixos, e do controlador de Klancar, pelo fato das velocidades de referência serem constantes para esse caso. Enquanto que os ganhos do controlador *backstepping* adaptativo, demonstram permanecerem praticamente constantes, com exceção do ganho da coordenada  $\theta$  que, nos instantes finais da simulação, apresenta comportamento crescente. Isso indica um princípio de instabilidade, comprovando o mal ajuste do controlador adaptativo para esta variável.

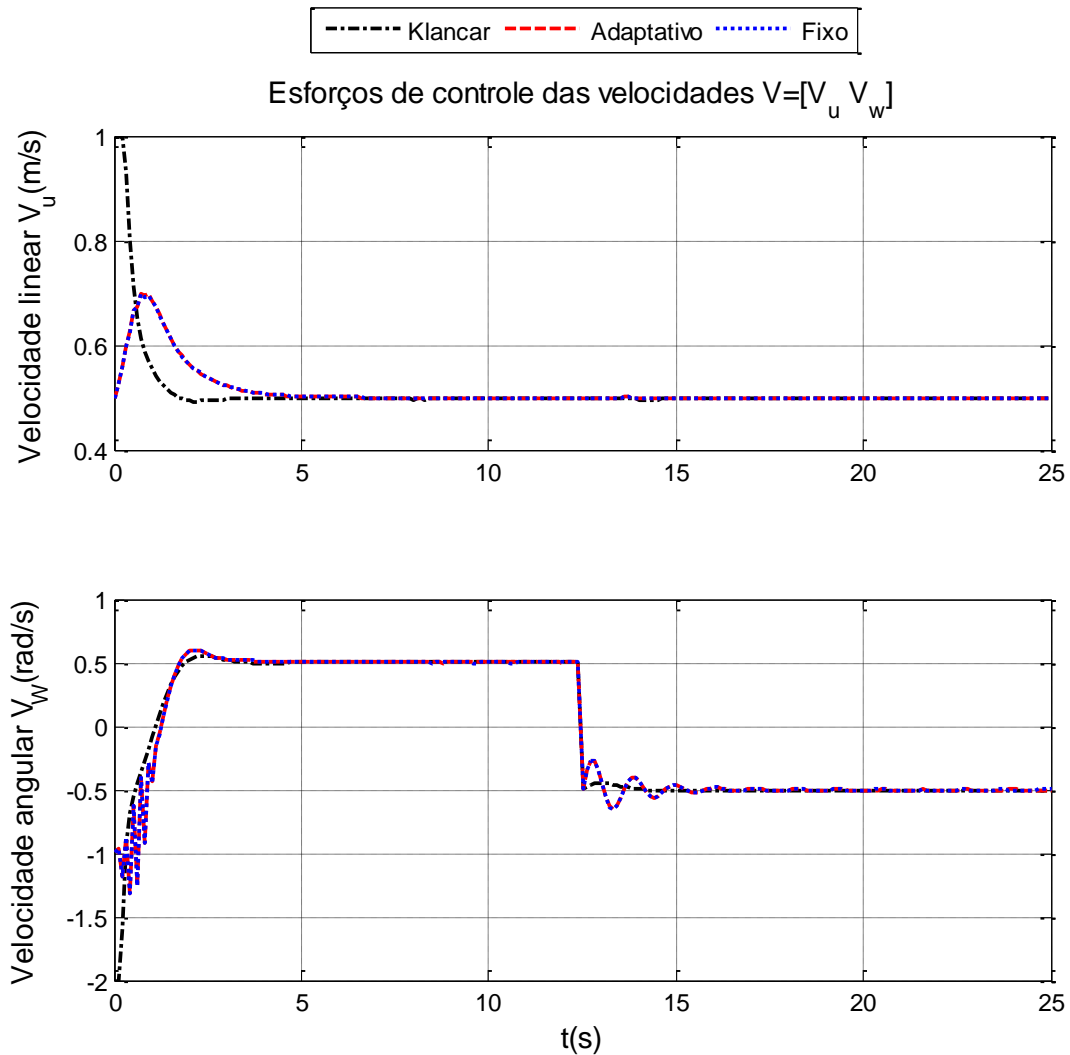
**Figura 4-3: Ganhos dos controladores ajustados conforme artigos.**



**Fonte: Autor.**

Atentando agora para o esforço de controle dos controladores mostrado na Figura 4-4, observa-se que o controlador de Klancar consegue, dentro do intervalo de 5s, convergir as velocidades linear e angular para valores próximos das referências,  $0,5\text{m/s}$  e  $0,5\text{rad/s}$  respectivamente, apresentando um pequeno *offset* da ordem de  $10^{-3}$ , demonstrando um bom desempenho de esforço de controle. Enquanto que os controladores *backstepping*, o clássico e o adaptativo, conseguem convergir a velocidade linear para a desejada de  $0,5\text{m/s}$  com um sobressinal inferior, ou seja, melhor que o apresentado pelo controlador de Klancar. Porém, a velocidade angular oscila bastante durante os transitórios. Com isso, percebe-se que os valores escolhidos no trabalho (MOHARERI, DHAOUADI e RAD, 2012) para os ganhos dos controladores não são adequados para o valores da planta adotados nesse trabalho.

**Figura 4-4: Esforço de controle das velocidades para ajustes conforme artigos.**



Fonte: Autor.

Portanto, o próximo passo será ajustar os controladores *backstepping* para a planta atual cujos os parâmetros foram especificados anteriormente, e manter os controlador de Klancar com os valores ajustados conforme em (KLANCAR, MATKO e BLAZIC, 2005).

#### 4.1 Ajuste dos controladores

Considerando as mesmas referências e condições iniciais escolhidas anteriormente, deu-se início ao ajuste dos ganhos do controlador *backstepping* de ganho fixo, que, salientando mais uma vez, serão utilizados como condição inicial dos ganhos do controlador *backstepping* adaptativo.

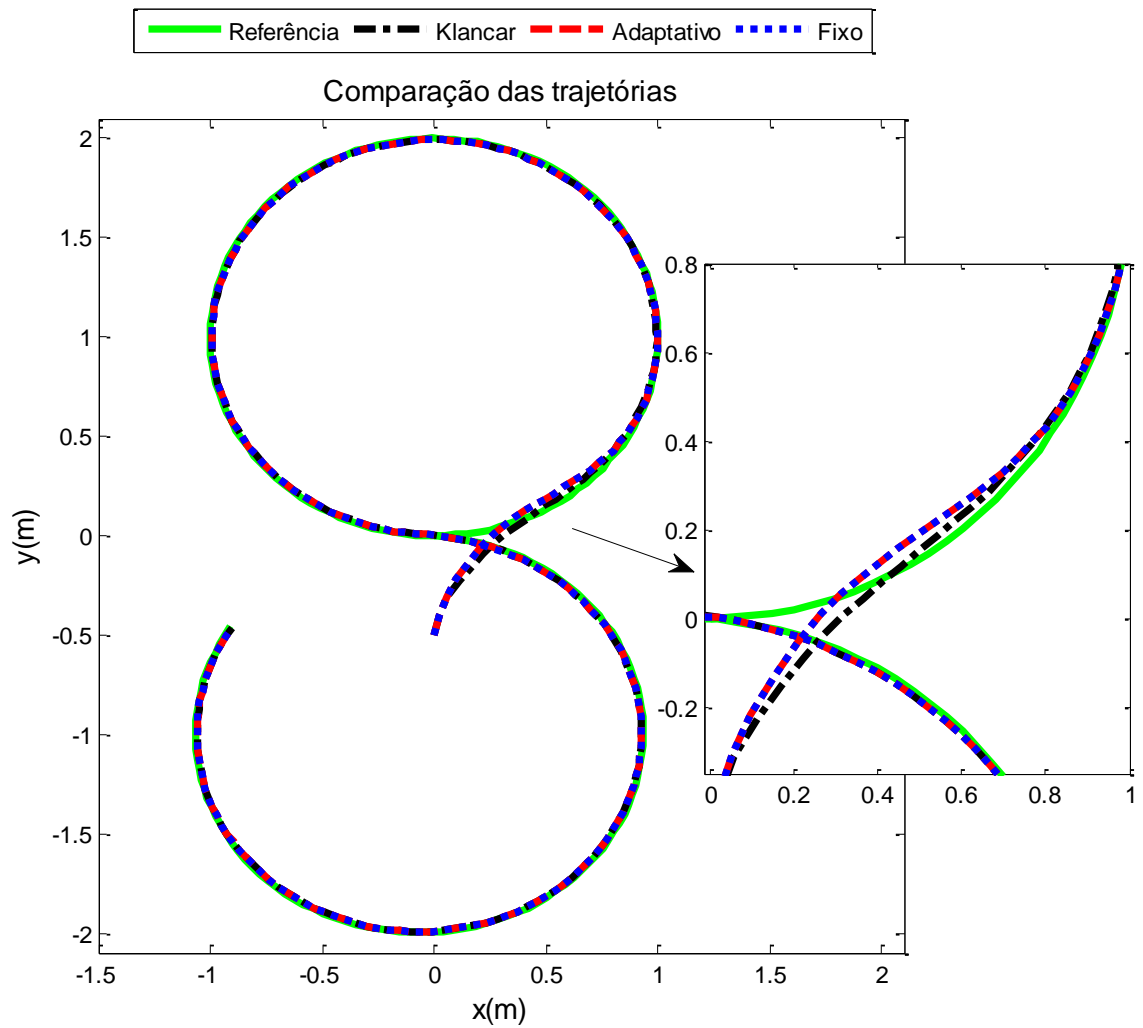
Após algumas simulações, o melhor resultado encontrado foi com os seguintes valores de ganho:

$$K_{x_0} = 10; K_{y_0} = 70; K_{\theta_0} = 20.$$

Com esse ajuste para os controladores *backstepping* clássico e para o adaptativo, e considerando o mesmo ajuste do controlador de Klancar  $\xi = 0,6$ ;  $\omega_n = 2$ ;  $g = 30$ , obteve-se as trajetórias resultantes mostradas sobrepostas na Figura 4-5.

Ao analisar este gráfico, entende-se que o desempenho é satisfatório, tendo em vista o bom rastreamento de trajetória apresentado, tendo o controlador de Klancar um ajuste para essa trajetória de referência ligeiramente melhor que os controladores *backstepping*.

**Figura 4-5: Trajetórias resultantes após a sintonia dos controladores.**

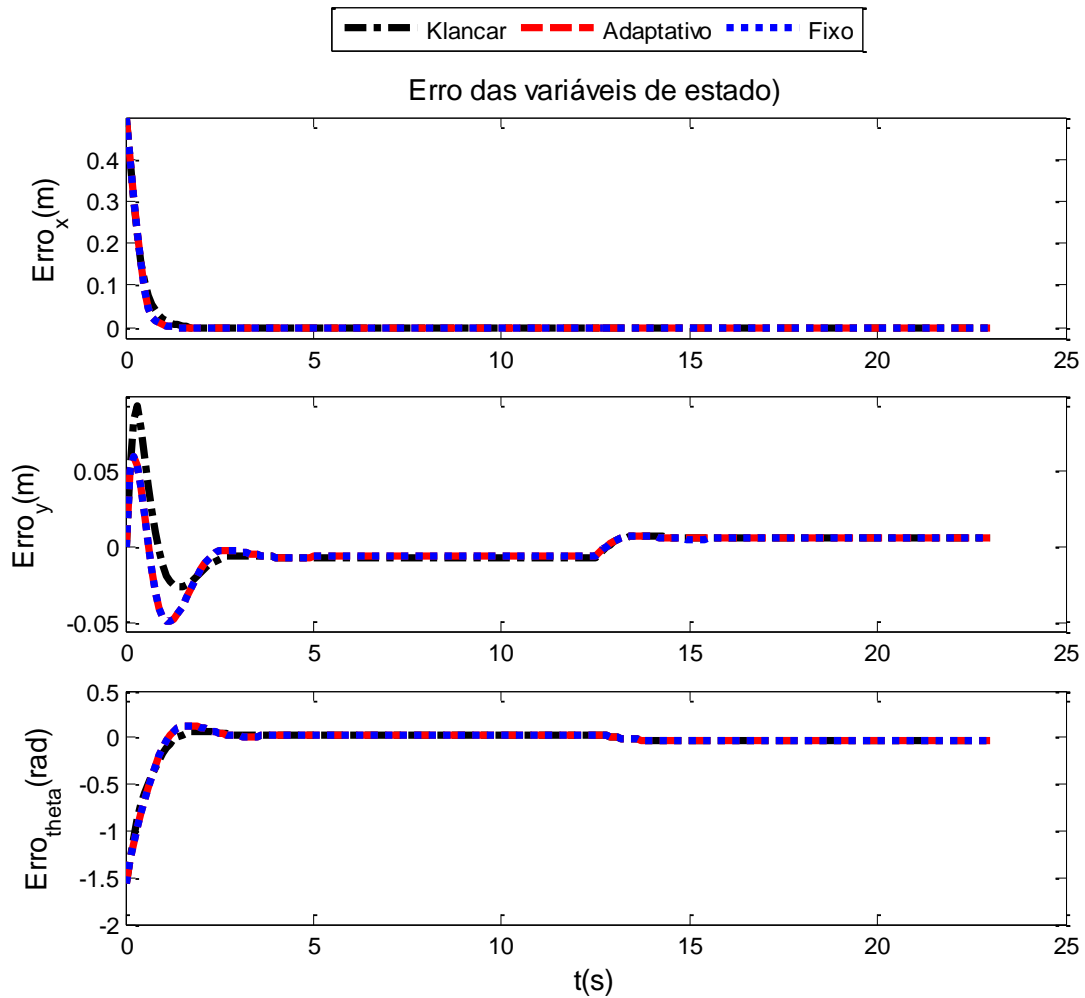


Fonte: Autor.



Analisando os erros das variáveis de estado na Figura 4-6, observa-se que os erros convergem, em um período inferior a 5s, para valores próximos à zero, na ordem de  $10^{-4}$ , demonstrando um bom desempenho dos controladores no regime permanente.

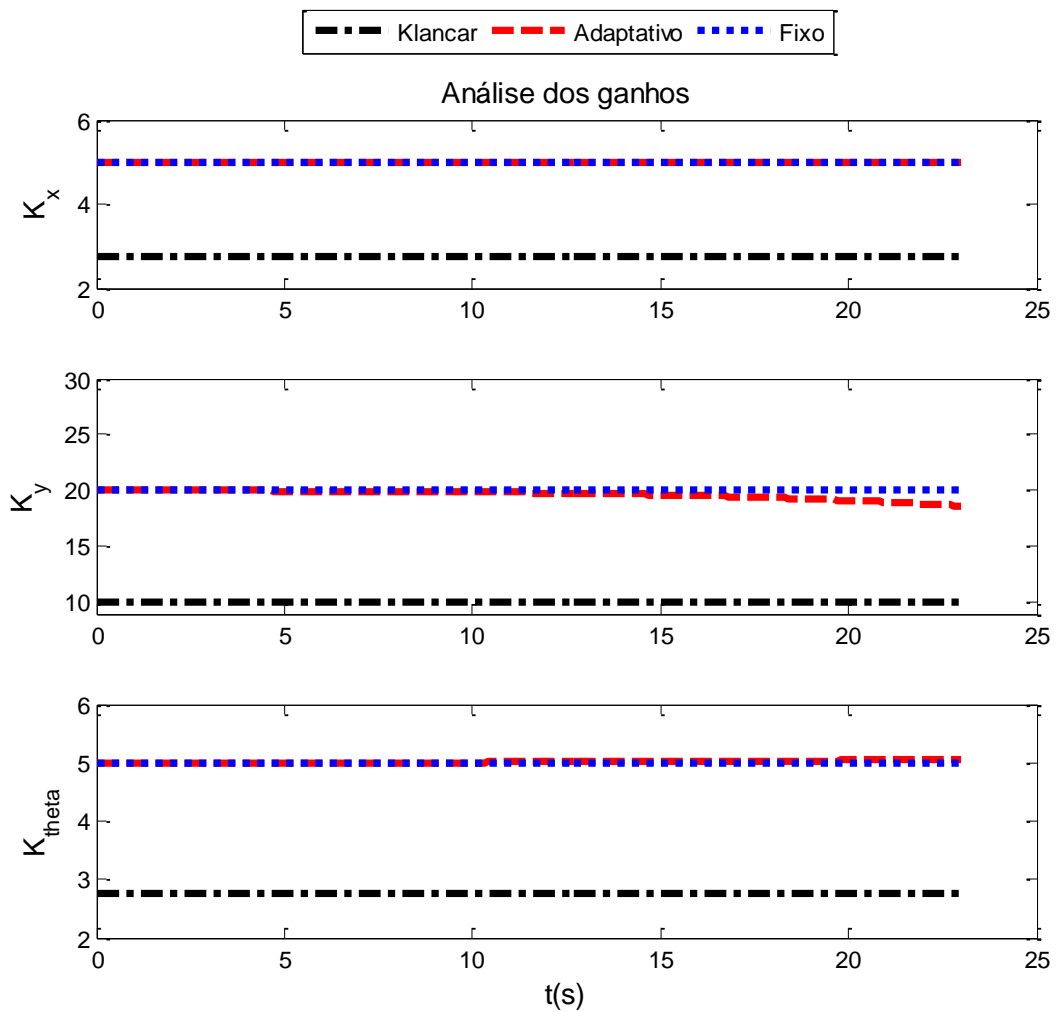
Figura 4-6: Erros de trajetória das variáveis de estado  $x$ ,  $y$  e  $\theta$  após sintonia dos controladores.



Fonte: Autor.

Na Figura 4-7, percebe-se a constância dos valores dos ganhos do controlador *backstepping* clássico e do controlador de Klancar, o que já era previsto conforme os mesmos motivos citados anteriormente. Enquanto que os ganhos do controlador *backstepping* adaptativo, demonstram permanecerem praticamente constantes, com exceção do ganho em  $y$  que tenta compensar o erro naquela coordenada nos instantes finais da simulação, apresenta comportamento decrescente.

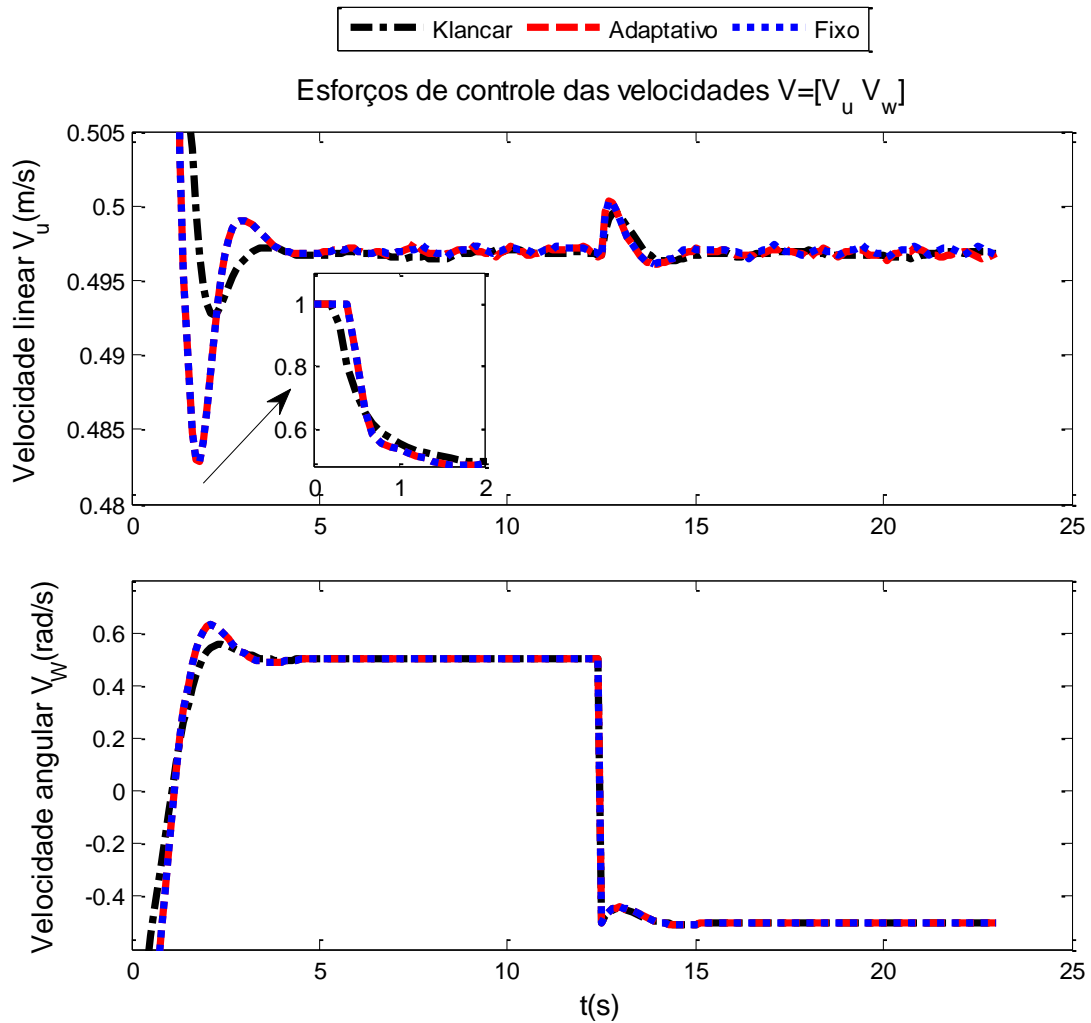
Figura 4-7: Ganhos dos controladores após sintonia dos controladores.



Fonte: Autor.

Atentando agora para os esforços de controle mostrados na Figura 4-8, observa-se que os controladores conseguem fazer com que a velocidade angular convirja para a desejada ( $0,5 \text{ rad/s}$ ) em um período próximo de  $5 \text{ s}$ . Enquanto que as velocidades lineares apresentaram comportamento oscilatório, embora tais valores sejam próximos aos de referência, na ordem de  $10^{-3}$ , demonstrando desempenho satisfatório.

**Figura 4-8: Esforço de controle das velocidades após sintonia dos controladores.**



Fonte: Autor.

Tal comportamento oscilatório da velocidade linear e o decaimento do ganho em  $y$  mostrado na Figura 4-7, poderiam ser corrigidos, porém os controladores *backstepping* demonstram ser bastante sensíveis às variações durante os ajustes. Isso pode ser uma desvantagem inicial no projeto do controlador *backstepping* adaptativo, entretanto, conforme será visto mais adiante, este controlador apresenta bom desempenho de autoajuste mesmo sem uma sintonia ótima.

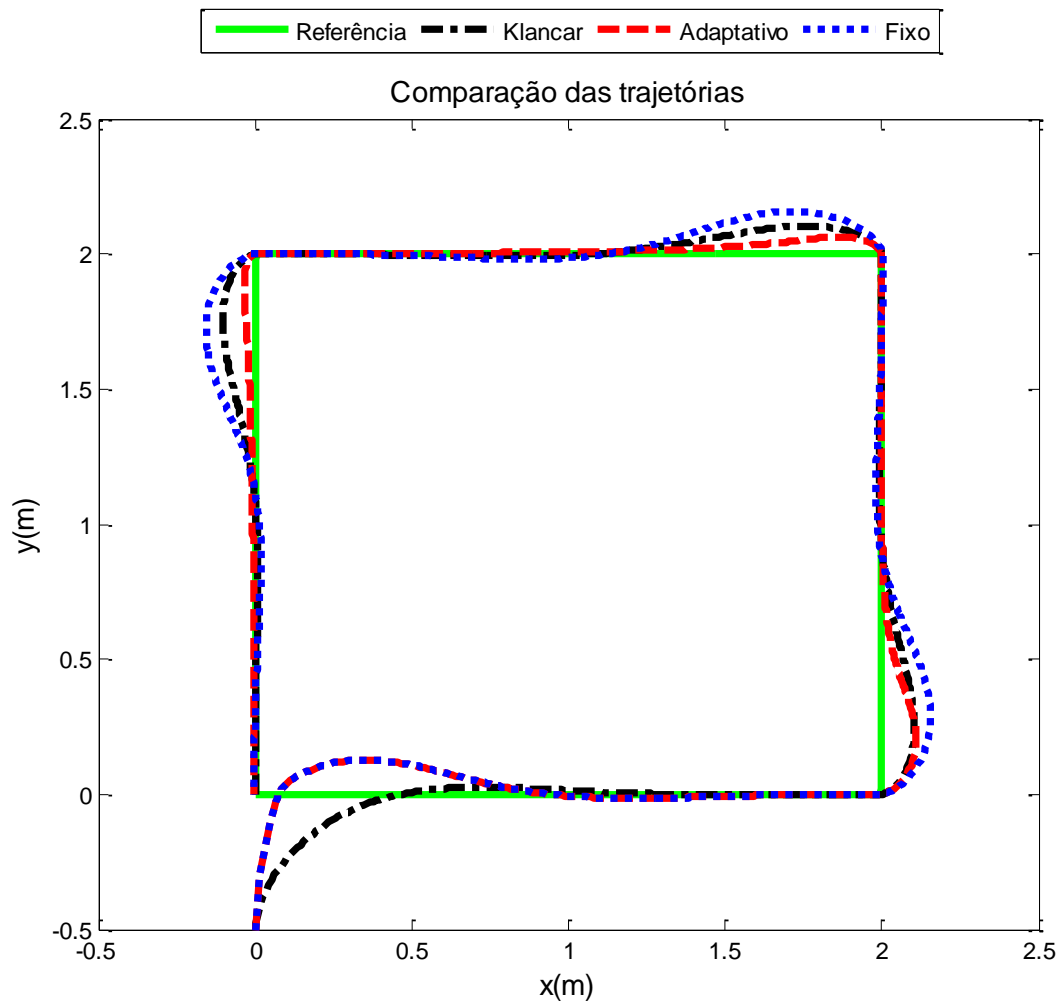
Então, com o objetivo de verificar o comportamento dos controladores para trajetórias diferentes da utilizada na sintonia, serão escolhidas outras duas trajetórias de referência conforme mostrado a seguir.

#### **4.2 Avaliação do desempenho dos controladores para diferentes trajetórias**

O desempenho dos dois controladores foi testado para duas diferentes trajetórias de referência, sendo inicialmente utilizada a em forma de quadrado, como mostrado na Figura 4-9. Analisando as trajetórias resultantes, percebe-se, rapidamente, a vantagem do controlador *backstepping* adaptativo proposto em ajustar-se à nova trajetória.

Verifica-se que o controlador inicia o rastreamento de trajetória com desempenho de semelhante ao do controlador *backstepping* clássico. Isso ocorre devido a suas condições iniciais estarem ajustadas conforme os valores utilizados neste controlador fixo.

Figura 4-9: Rastreamento para a trajetória de um quadrado de lado dois (2m).

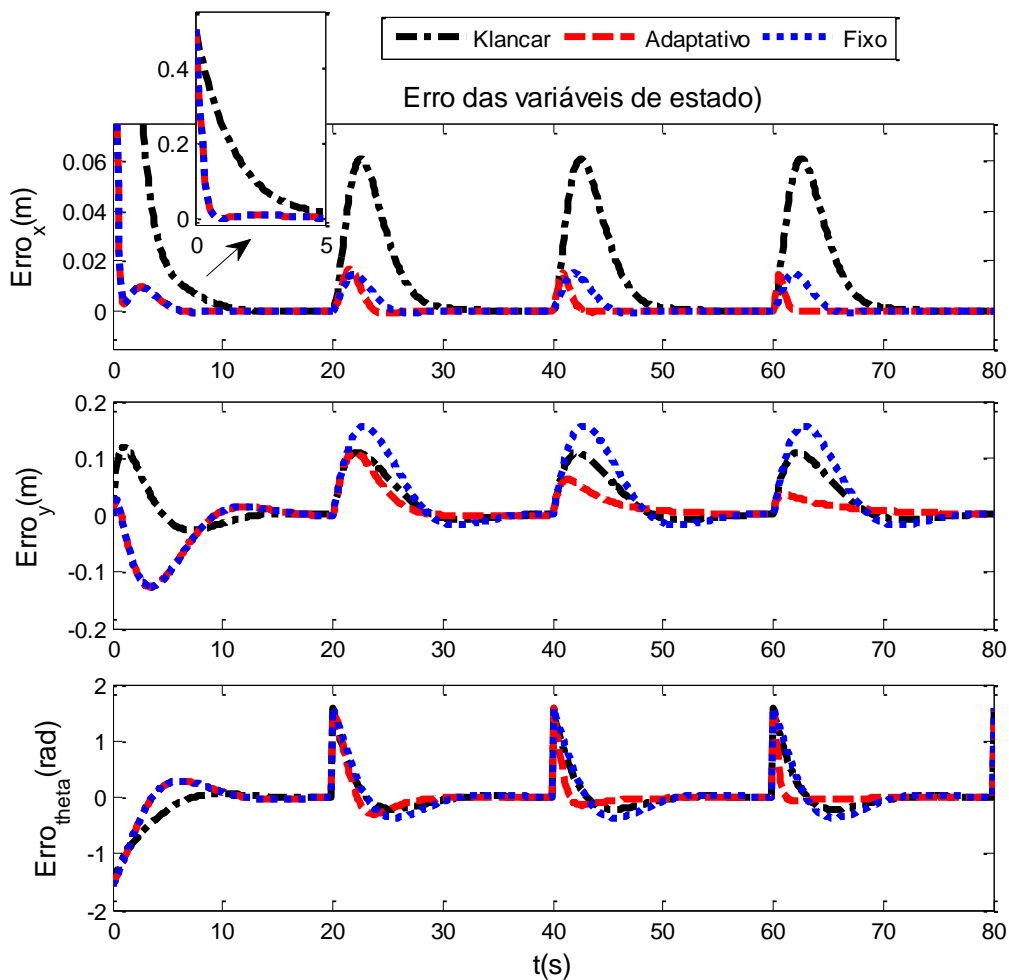


Fonte: Autor.

Analisando o desempenho do controlador de Klancar, percebe-se que este conseguiu rastrear a trajetória de referência mais rapidamente que os controladores *backstepping*.

Porém, ao verificar o gráfico dos erros mostrado na Figura 4-10, observa-se que os controladores *backstepping*, tanto o fixo como o adaptativo, apresentaram melhor convergência, corrigindo os erros de forma mais rápida que o controlador de Klancar, principalmente o da coordenada  $x$ . Percebe-se, ainda, que o controlador adaptativo corrige os erros das coordenadas de trajetória de forma mais rápida a medida que o robô se move ao longo da trajetória.

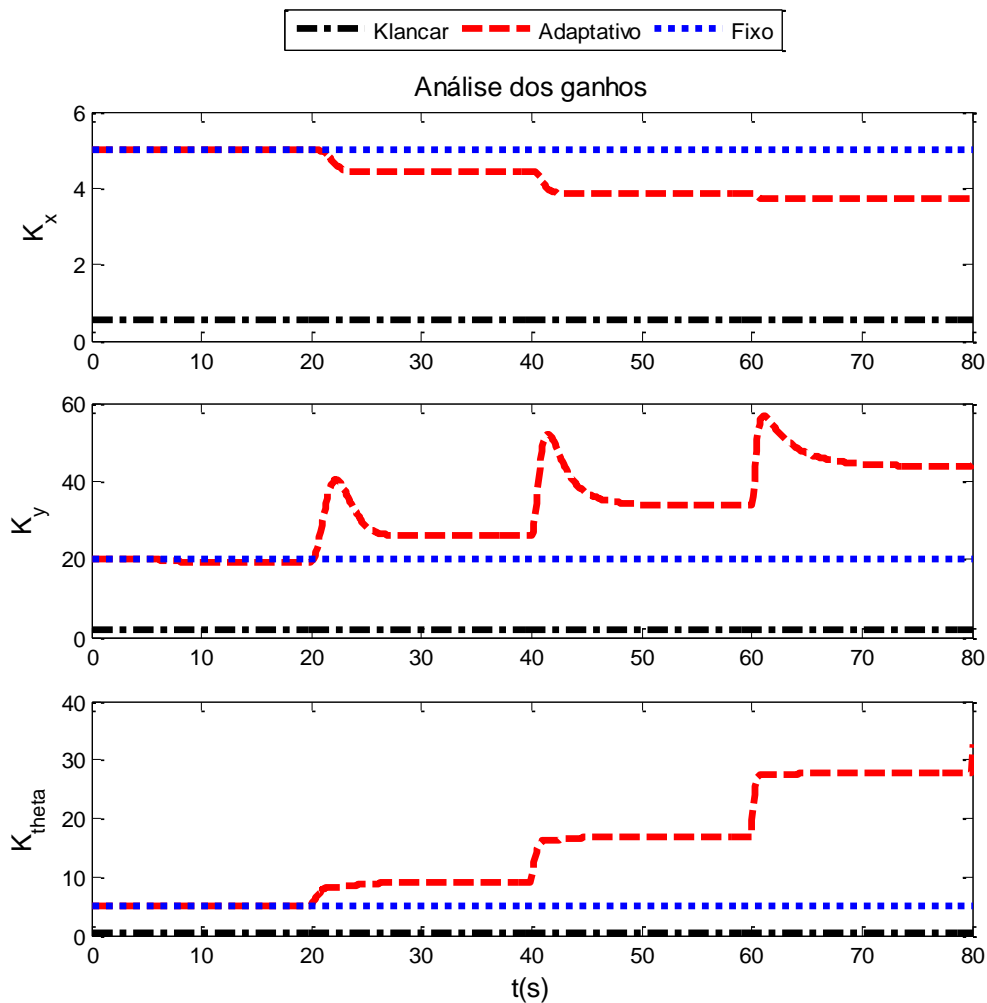
**Figura 4-10:** Erros de trajetória das variáveis de estado  $x$ ,  $y$  e  $\theta$  para a trajetória de um quadrado.



Fonte: Autor.

Na Figura 4-11, é fácil verificar a capacidade de ajuste dos ganhos do controlador adaptativo a cada curva da trajetória quadrada. Conforme demonstrado matematicamente no tópico 3.3, esse método de controle adaptativo faz o ajuste *online* do ganho para diferentes tipos de percurso de referência, tendo um desempenho de rastreamento mais preciso a medida que o robô se move ao longo de qualquer trajetória. Enquanto que os ganhos do controlador com ganho fixo são ajustados para um único tipo de trajetória, não agindo de forma adequada para outras trajetórias. Já o controlador de Klancar atua de forma constante, haja visto o ajuste de seus ganhos por agendamento (*gain scheduling*) leva em consideração as velocidades linear e angular desejadas, as quais são constantes para esse caso, conforme verificado na Figura 4-11.

**Figura 4-11: Ganhos dos controladores para a trajetória de um quadrado.**

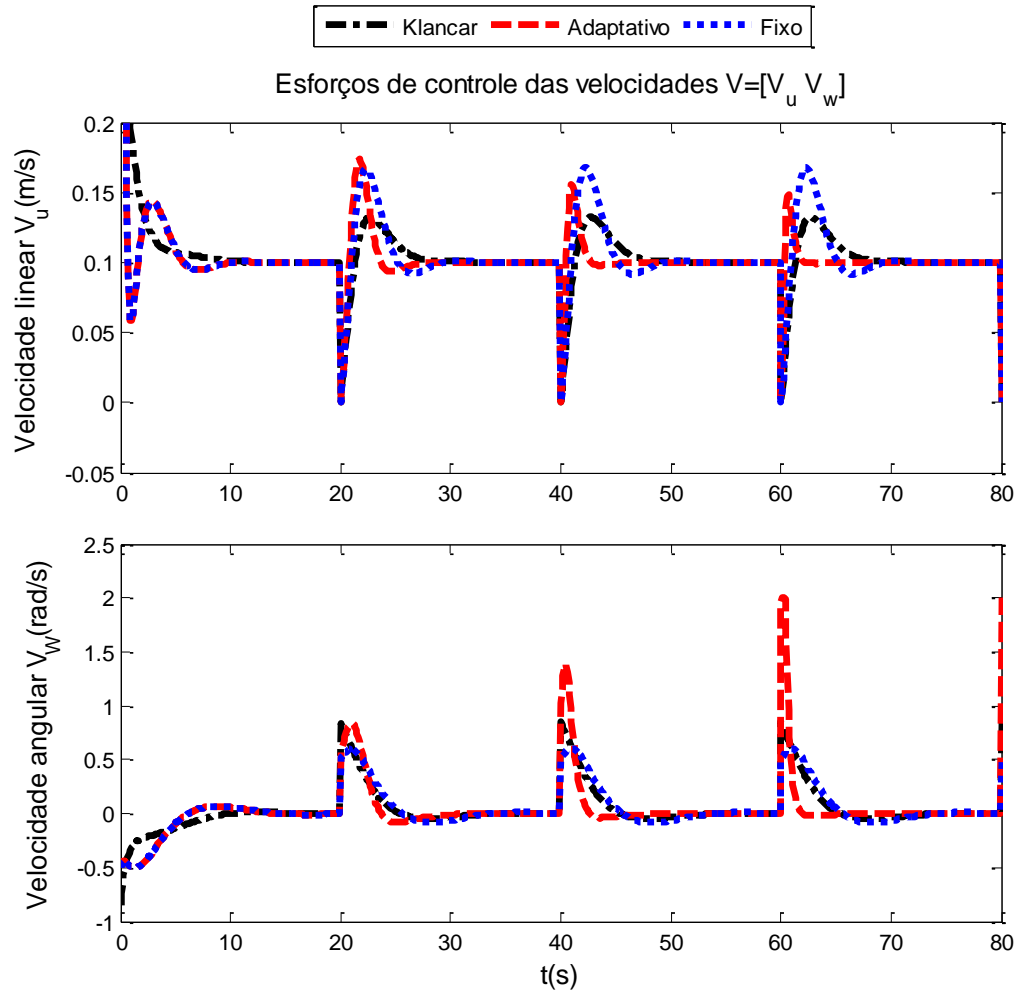


**Fonte: Autor.**

Atentando para os esforços de controle mostrados na Figura 4-12, observa-se que os controladores conseguem fazer com que as velocidade linear e angular converjam para a

desejada ( $0,5\text{m/s}$  e  $0,5\text{ rad/s}$  respectivamente) em um período próximo de  $10\text{s}$ . Percebe-se, ainda, que o sinal de controle de velocidade linear e angular enviado pelo controlador *backstepping* adaptativo oscila menos a medida que o robô percorre a trajetória.

**Figura 4-12: Esforço de controle das velocidades para a trajetória de um quadrado.**

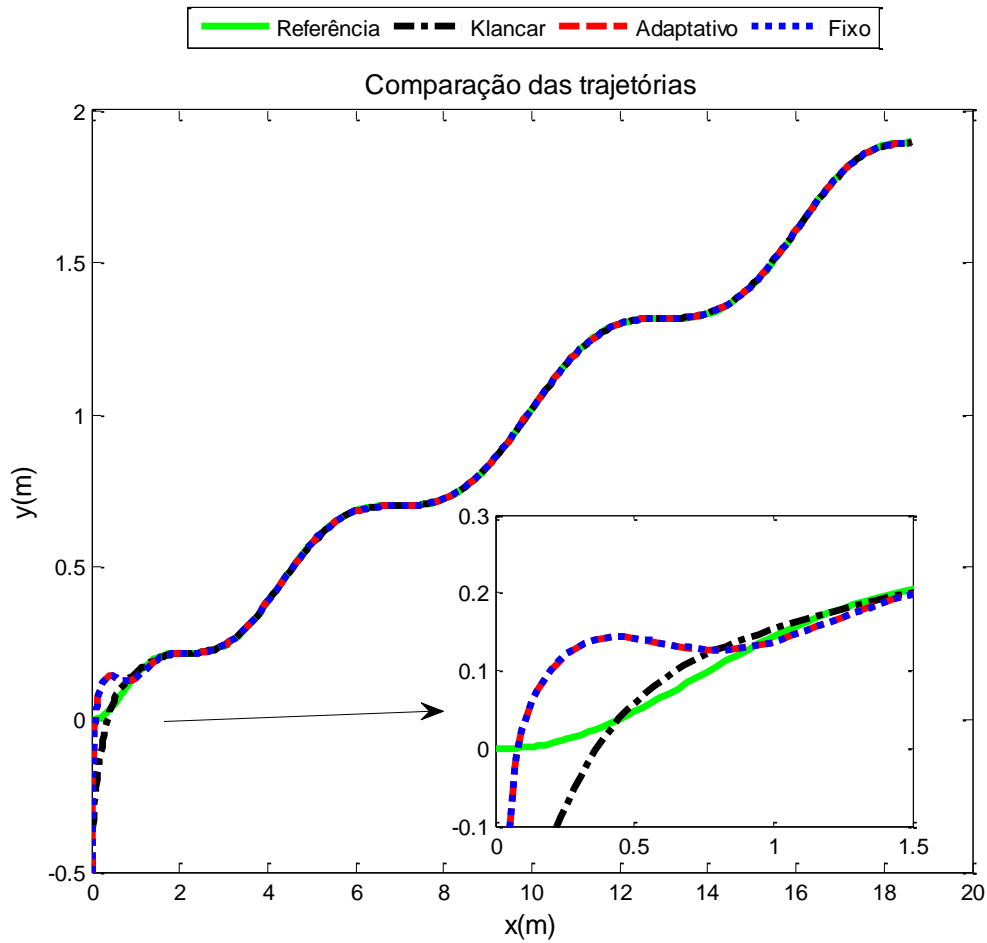


Fonte: Autor.



Testando o desempenho dos controladores agora para uma trajetória em forma de senóide, obteve-se os como mostrado na Figura 4-13. Verifica-se, mais uma vez, o rastreamento inicial semelhante entre os controladores *backstepping*, cujo motivo já foi explicado anteriormente.

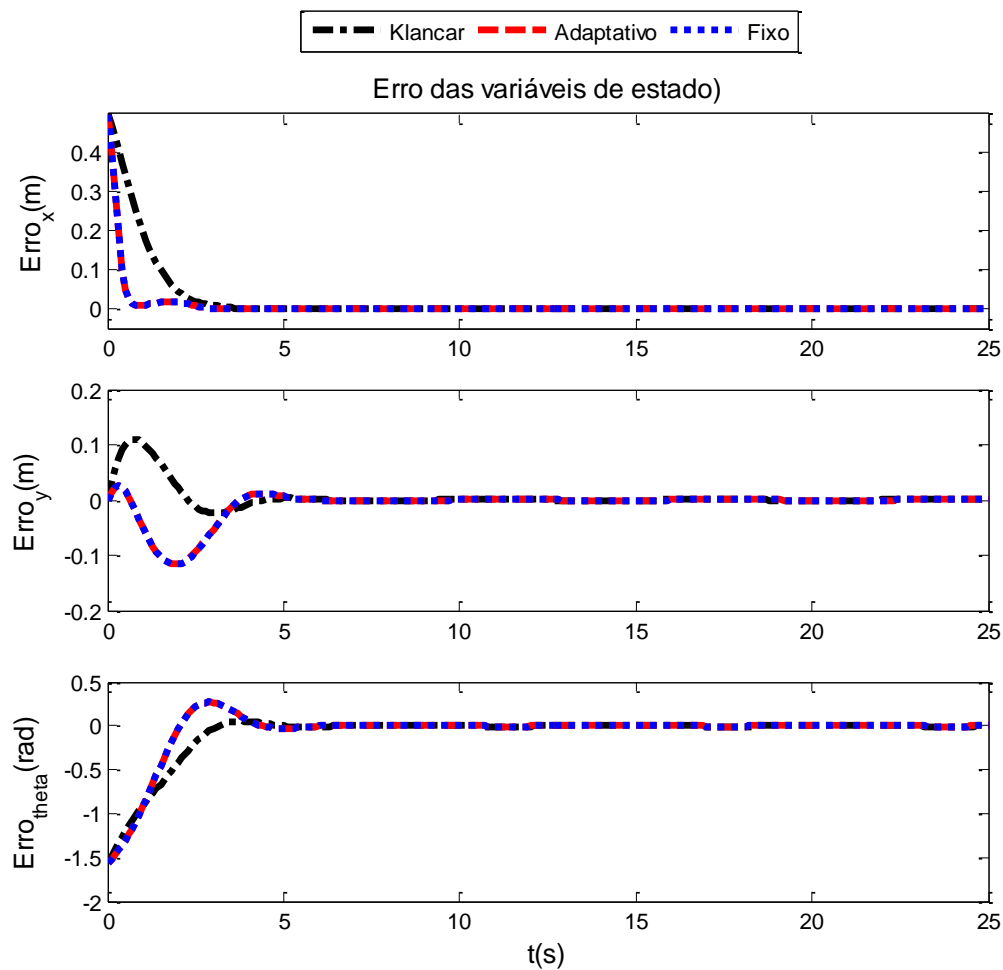
**Figura 4-13: Rastreamento para a trajetória de uma senóide.**



Fonte: Autor.

Analisando as trajetórias resultantes, percebe-se que os controladores conseguem rastrear satisfatoriamente a trajetória de referência, apresentando o controlador de Klancar uma menor oscilação inicial. Porém, ao verificar o gráfico dos erros mostrado na Figura 4-14, da mesma forma que para a trajetória anterior, observa-se que os controladores *backstepping*, tanto o fixo como o adaptativo, apresentaram melhor convergência, corrigindo os erros de forma mais rápida que o controlador de Klancar, principalmente o da coordenada  $x$ .

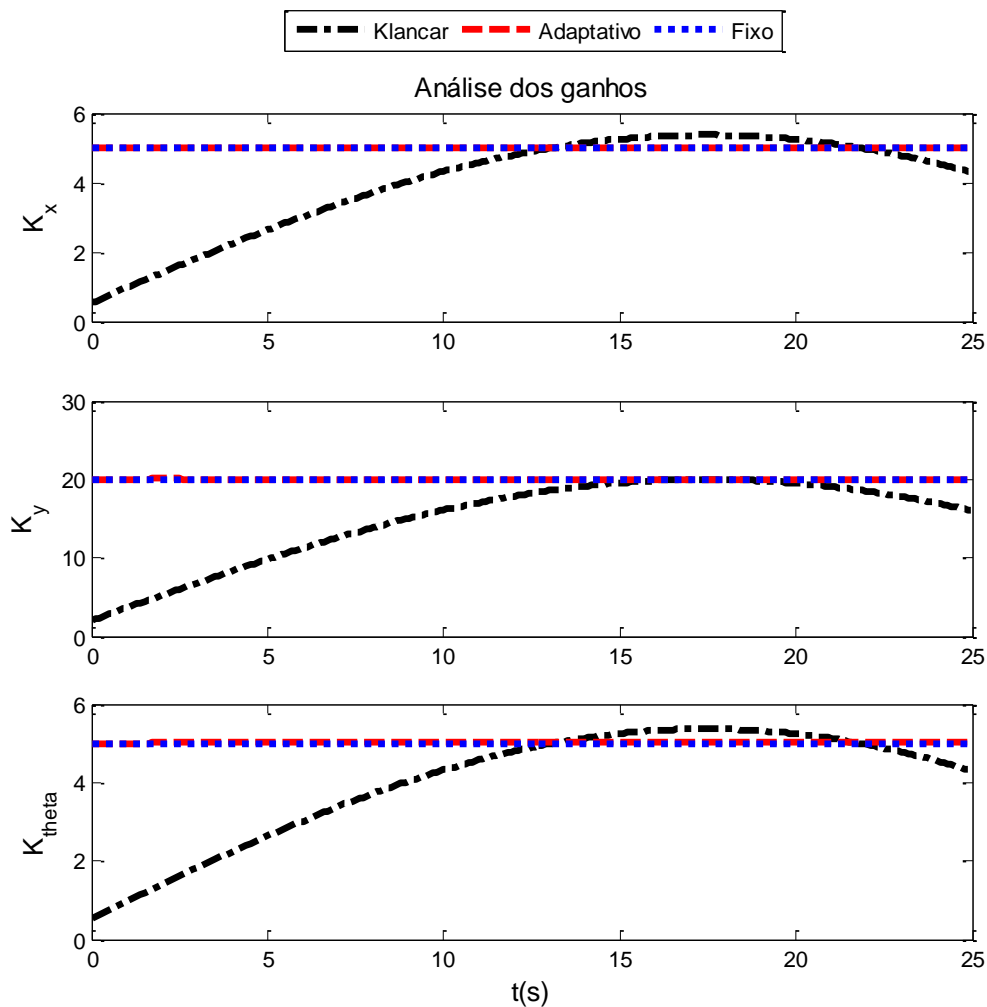
**Figura 4-14:** Erros de trajetória das variáveis de estado  $x$ ,  $y$  e  $\theta$  para a trajetória de uma senóide.



Fonte: Autor.

Na Figura 4-15, percebe-se, que o ganho do controlador *backstepping* adaptativo praticamente não varia, apresentando comportamento semelhante ao controlador *backstepping* clássico, enquanto que o controlador de Klancar apresenta uma variação de ganhos. Isso ocorre devido ao método levar em consideração nos ajustes dos ganhos as velocidades linear e angular desejadas, as quais são variáveis para a trajetória senoidal de referência, cujas amplitudes são de  $0,9 \text{ m/s}$  para a velocidade linear e de  $0,1 \text{ rad/s}$  para a velocidade angular, a uma frequência de  $1 \text{ rad/s}$  e  $0,09 \text{ rad/s}$ , respectivamente. Os esforços de controle dos controladores para tais velocidades de referência, podem ser visualizados na Figura 4-16.

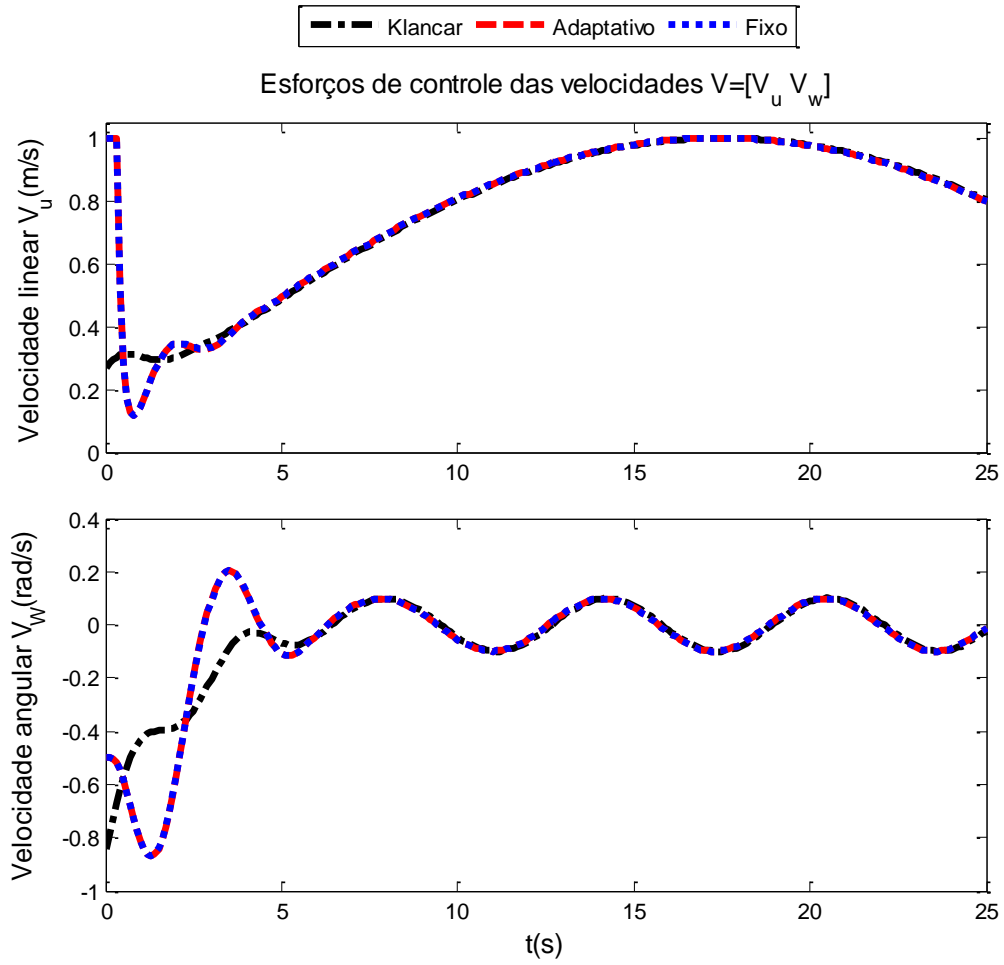
**Figura 4-15: Ganhos dos controladores para a trajetória de uma senóide.**



**Fonte: Autor.**

Percebe-se, ao analisar a Figura 4-16, que os controladores conseguem fazer com que o sistema siga a trajetória de referência. Os desempenhos são satisfatórios, haja visto a pouca oscilação no transitório e a boa convergência no regime permanente.

Figura 4-16: Esforço de controle para trajetória de uma senóide.



Fonte: Autor.

E, por fim, com o objetivo de verificar e comparar a robustez dos controladores, serão inseridos no sistema perturbações na cinemática e na dinâmica do sistema.

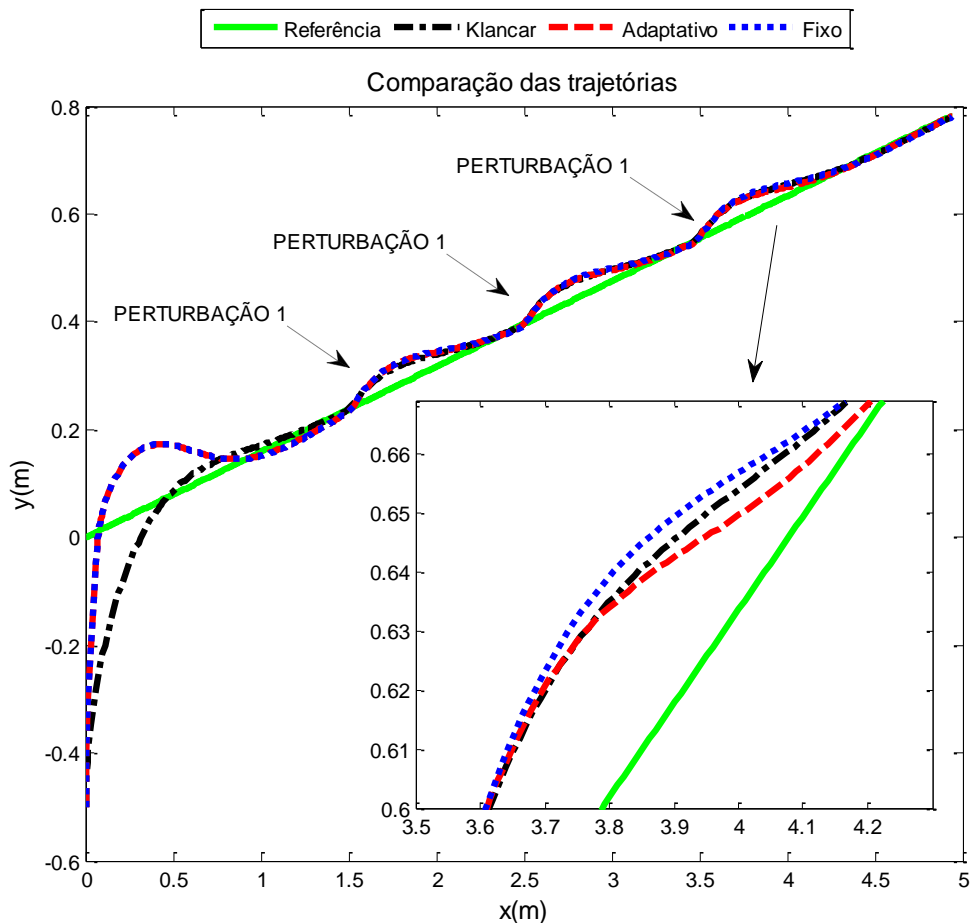
### 4.3 Avaliação do desempenho dos controladores contra perturbações e distúrbios

A fim de comparar o desempenho contra o rejeito de perturbações e ruídos dos três tipos de controladores em análise, foi simulada uma trajetória de referência de uma reta inclinada e aplicou-se três perturbações no sistema.

Tais perturbações foram aplicadas no motor esquerdo do robô, simulando um travamento, que irá reduzir a velocidade em  $2\text{ m/s}$  durante  $1\text{ s}$  em três instantes:

Como pode ser observado na Figura 4-17, a rejeição de distúrbios e robustez dos três controladores analisados. Contudo, o método com ajuste adaptativo dos ganhos apresenta melhor rejeição do terceiro distúrbio, conforme destacado na Figura 4-17.

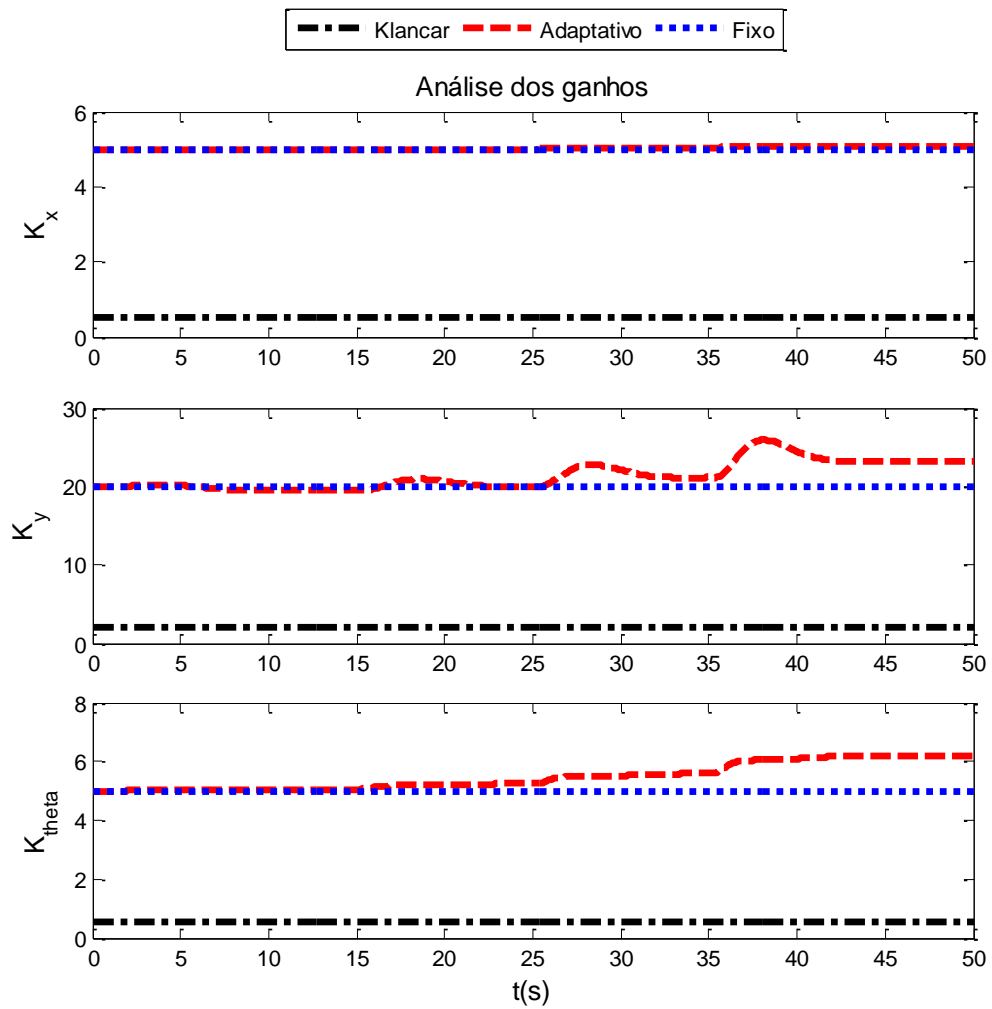
Figura 4-17: Rastreamento de uma trajetória com distúrbios no sistema.



Fonte: Autor.

A Figura 4-18 mostra a capacidade de ajuste dos ganhos do controlador backstepping adaptativo para a situação com os distúrbios aplicados. Isso mostra a superioridade de robustez desse controlador com relação ao *backstepping* clássico.

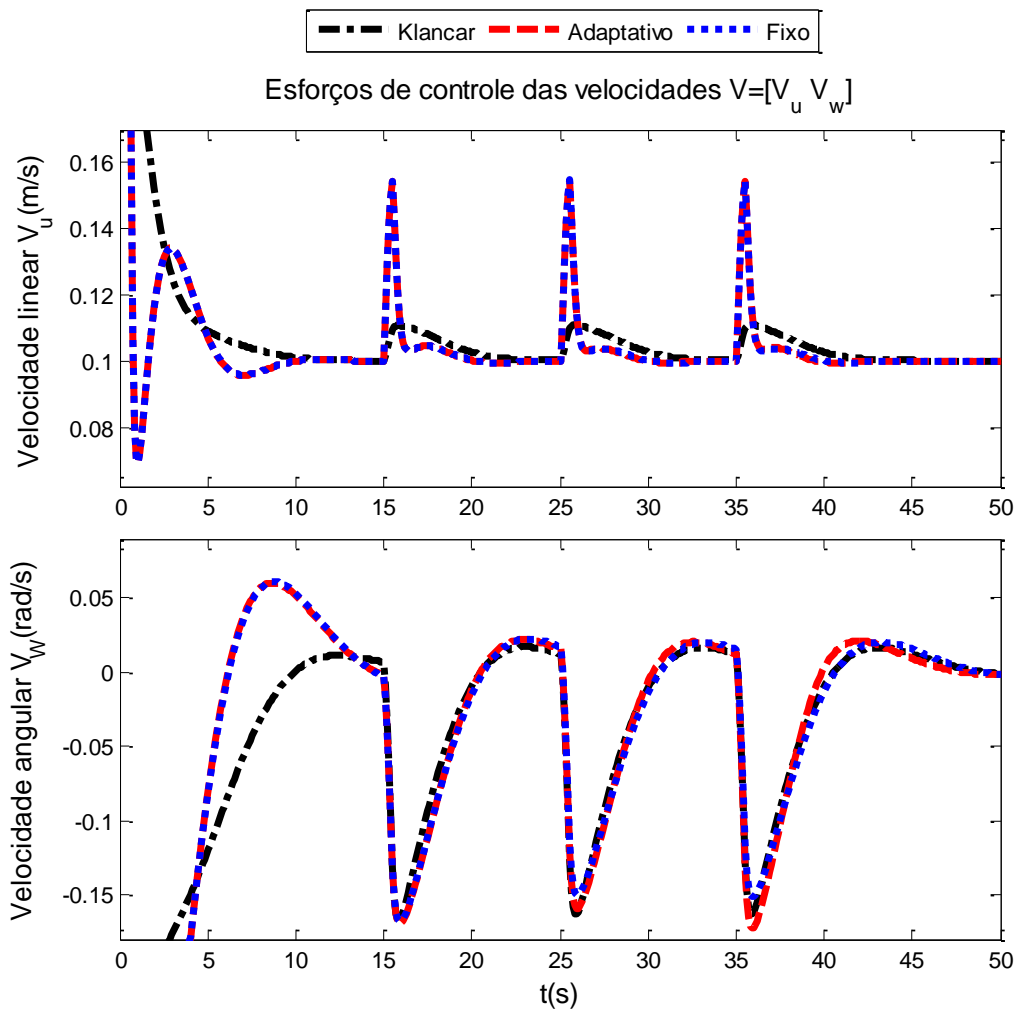
**Figura 4-18: Ganhos dos controladores na presença de distúrbios no sistema.**



Fonte: Autor.

Os esforços de controle das velocidades linear e angular são evidenciados na Figura 4-19, onde percebe-se que os controladores *backstepping* exigem mais esforço dos motores do robô que o controlador de Klancar. Contudo, eles conseguem convergir para a velocidade de referência angular e linear de forma mais rápida que o controlador de Klancar. Percebe-se ainda que o esforço do controlador adaptativo tende a diminuir com o tempo devido a sua capacidade de autoajuste.

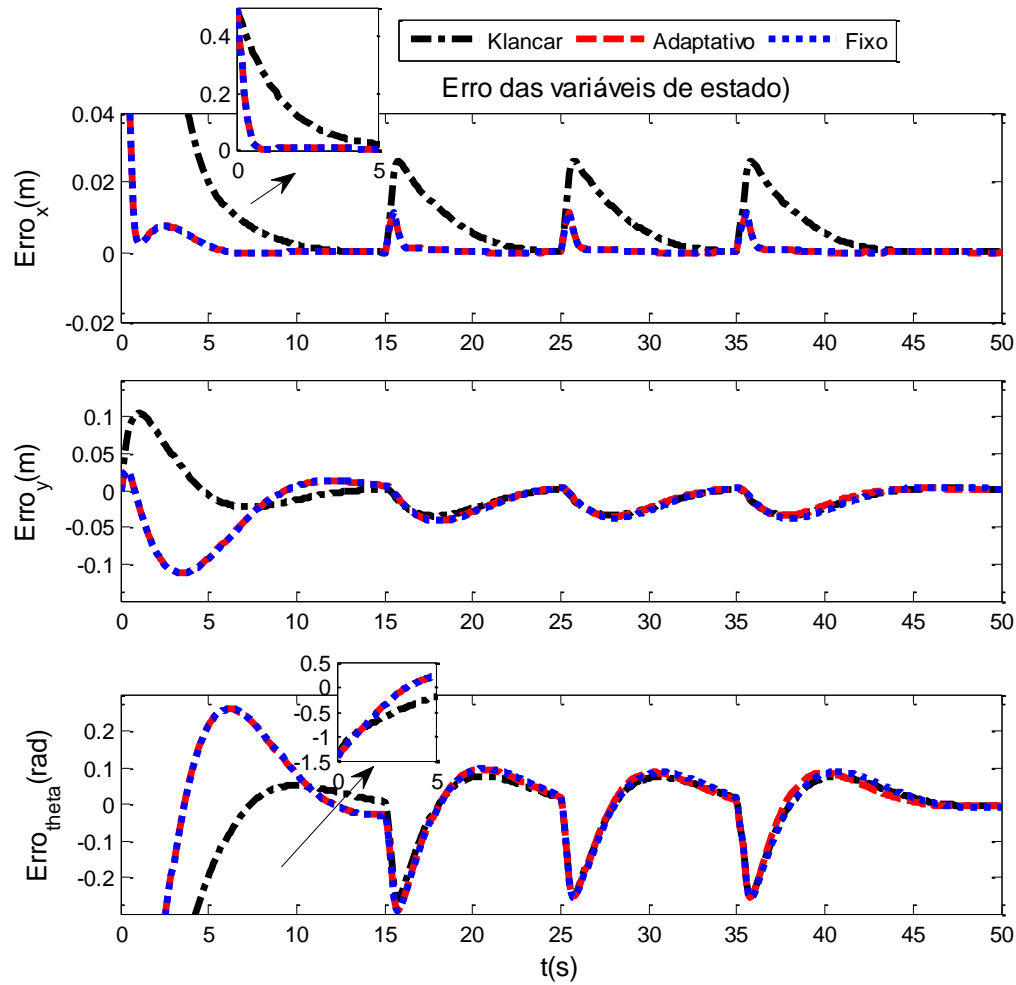
Figura 4-19: Esforço de controle para distúrbios no sistema.



Fonte: Autor.

Analisando o gráfico dos erros mostrado na Figura 4-20, observa-se que os controladores *backstepping*, tanto o fixo como o adaptativo, apresentaram melhor convergência, corrigindo os erros de forma mais rápida que o controlador de Klancar, principalmente o da coordenada  $x$ .

Figura 4-20: Erro.



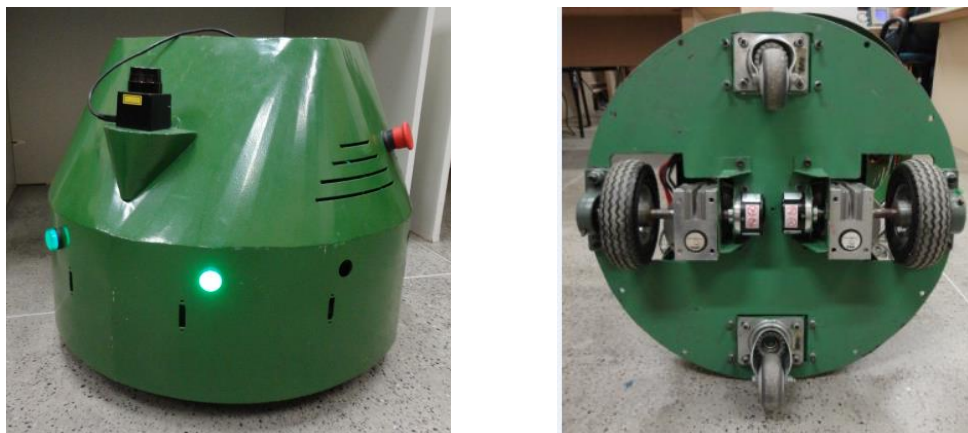
Fonte: Autor.



## 5 RESULTADOS EXPERIMENTAIS

Para a realização dos ensaios experimentais das estratégias de controle foi utilizado um robô móvel com duas rodas desenvolvido no departamento de engenharia elétrica da Universidade Federal do Ceará (UFC), conforme ilustrado na Figura 5-1. As duas rodas maiores possuem acionamento diferencial através de conjunto motor-reductor e encoders, as duas rodas menores possuem giro livre e são responsáveis por o equilíbrio do robô no plano.

**Figura 5-1: Robô móvel com duas rodas desenvolvido na UFC.**



Fonte: Autor.

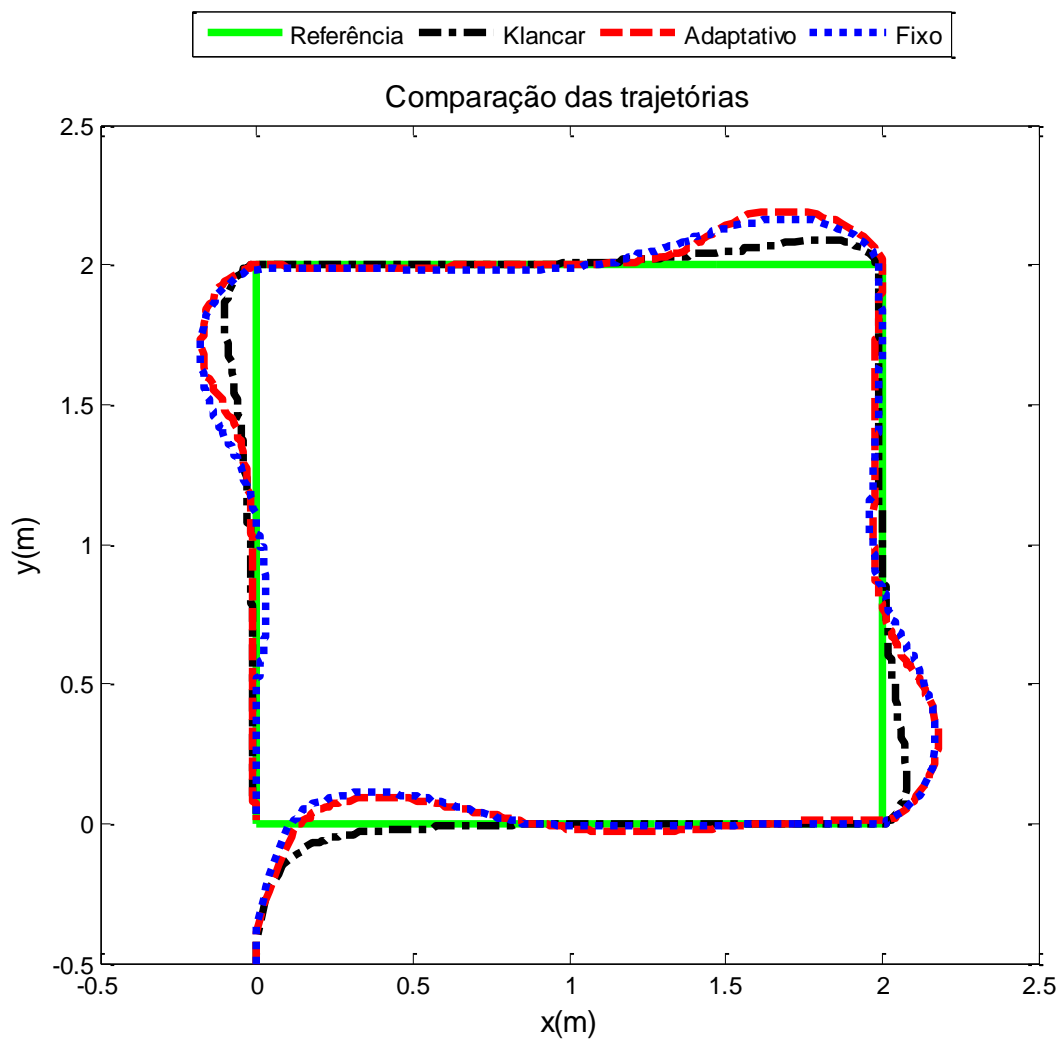
O controle de velocidade dos dois motores DC é realizado através de um driver de potência que possui malhas de controle PID internas. A realimentação de velocidade é obtida através de encoders incrementais acoplados aos eixos dos motores. Por outro lado, as estratégias de controle para seguimento de trajetórias foi implementada em uma placa FRDM K64F, a qual também é responsável pela comunicação com um sistema supervisor. A programação do *firmware* foi realizada em linguagem *c* utilizando a máquina DSP do processador ARM. O intervalo de amostragem utilizado na implementação real foi de 100 ms.

### 5.1 Ensaio para trajetória de referência de um oito

Determinou-se uma trajetória de referência na forma de um “quadrado” com as mesmas especificações do caso simulado. Considerou-se que o robô estará em uma condição de erro de estado inicial, ou seja, não partindo da origem da trajetória de referência, e sim do ponto  $(X_0, Y_0) = (0, 0, -0,5)$ , posicionado a um ângulo  $\theta = 90$  graus em relação ao eixo de referência. A velocidade linear de referência foi de  $0,1m/s$  e a velocidade angular de referência nula.

A Figura 5-2 apresenta as respostas dos três controladores para a trajetória de referência. Percebe-se que para o ajuste adotado o controlador Klancar resultou em uma curva com sobre sinal reduzido quando comparado com os demais. Contudo, percebe-se que o controlador adaptativo se aproxima da resposta do controlador Klancar a medida que a trajetória é percorrida, o que mostra a adaptação do controlador atuando no sentido de redução dos erros.

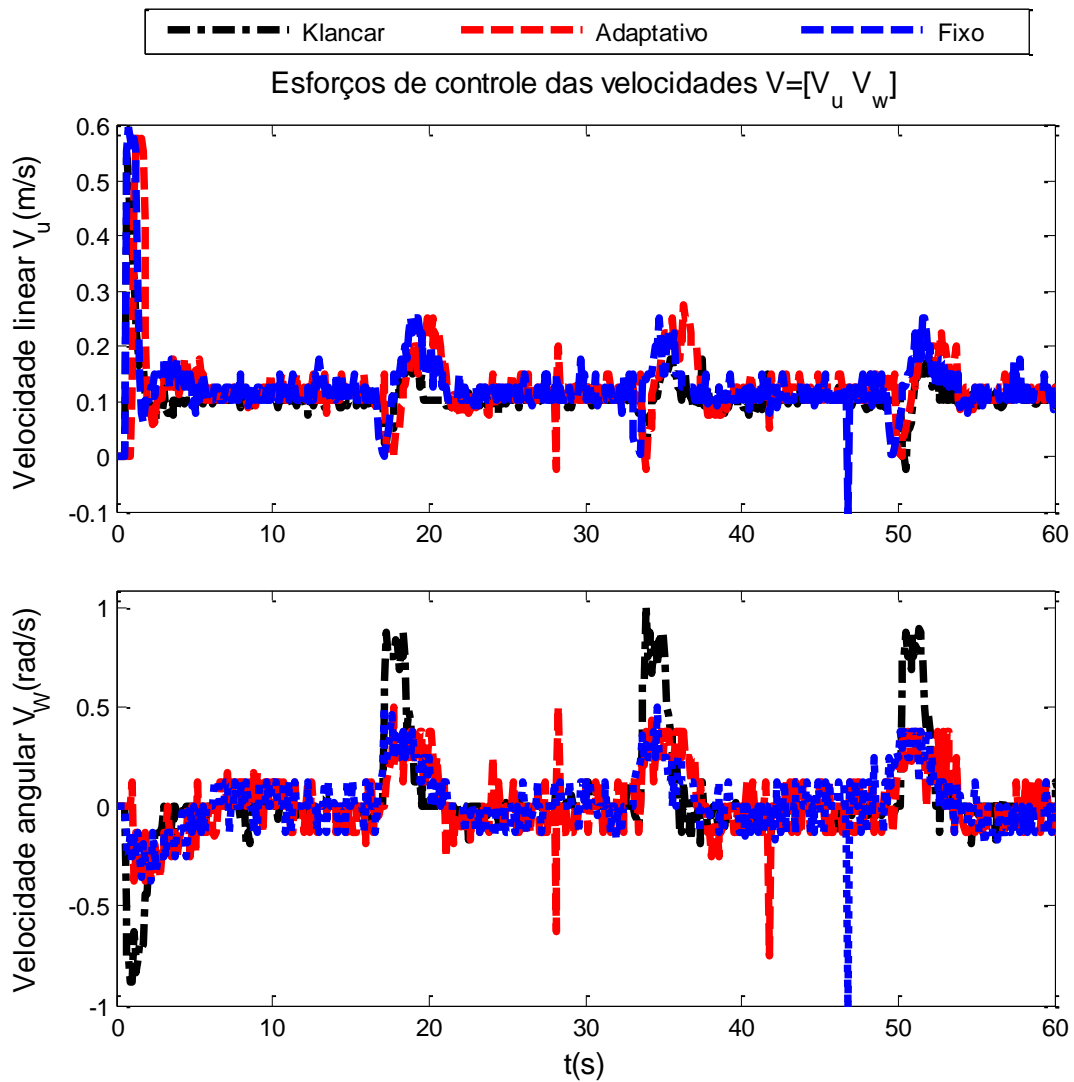
**Figura 5-2: Seguimento do robô real para uma trajetória na forma de um quadrado.**



Fonte: Autor.

A Figura 5-3 apresenta os respectivos valores de esforço de controle para a trajetória avaliada na prática. Nota-se que o controlador Klancar apresenta um esforço de controle superior quando comparado com os demais controladores.

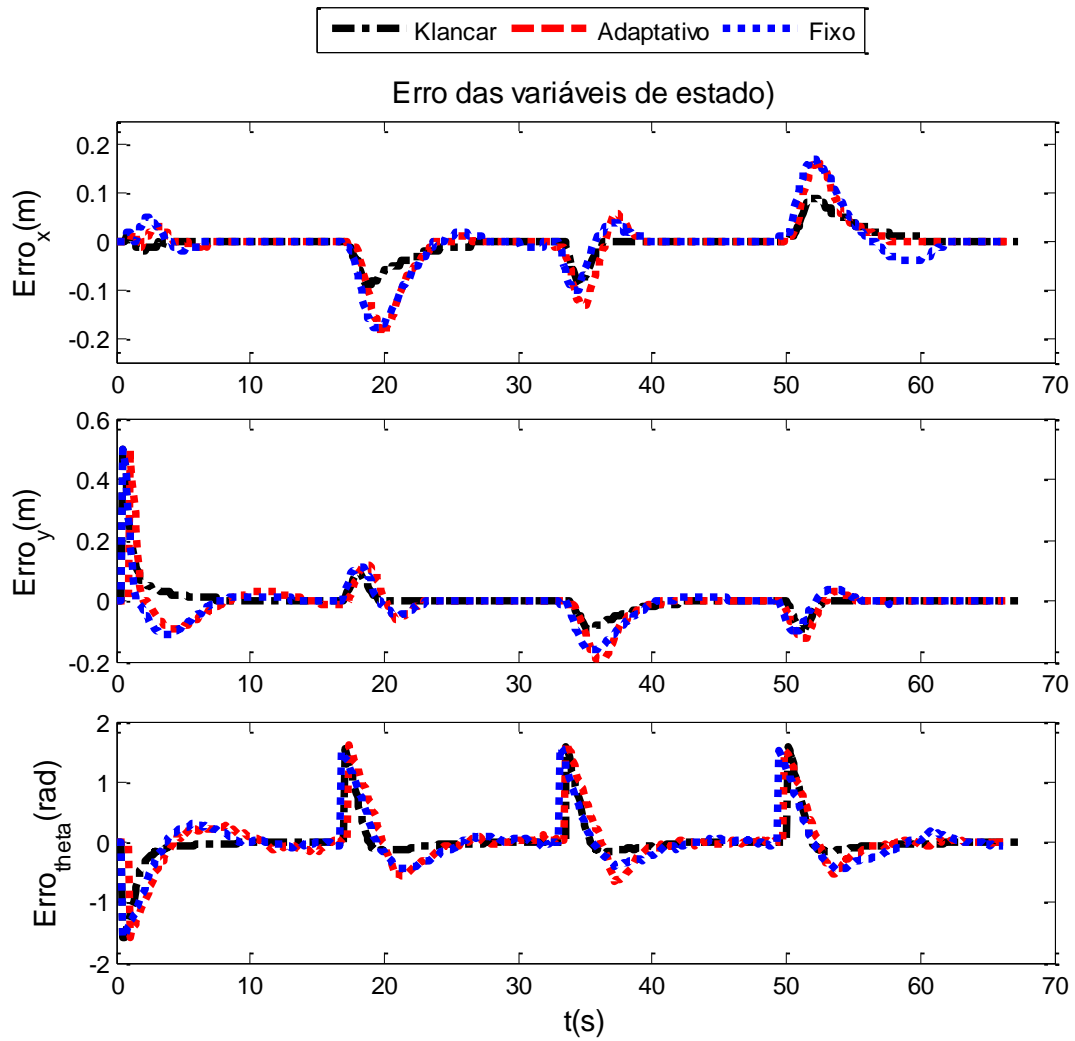
Figura 5-3: Trajetória de referência usada para ajuste dos controladores.



Fonte: Autor.

A Figura 5-4 apresenta os respectivos valores de erros nas variáveis  $x$ ,  $y$  e  $\theta$ .

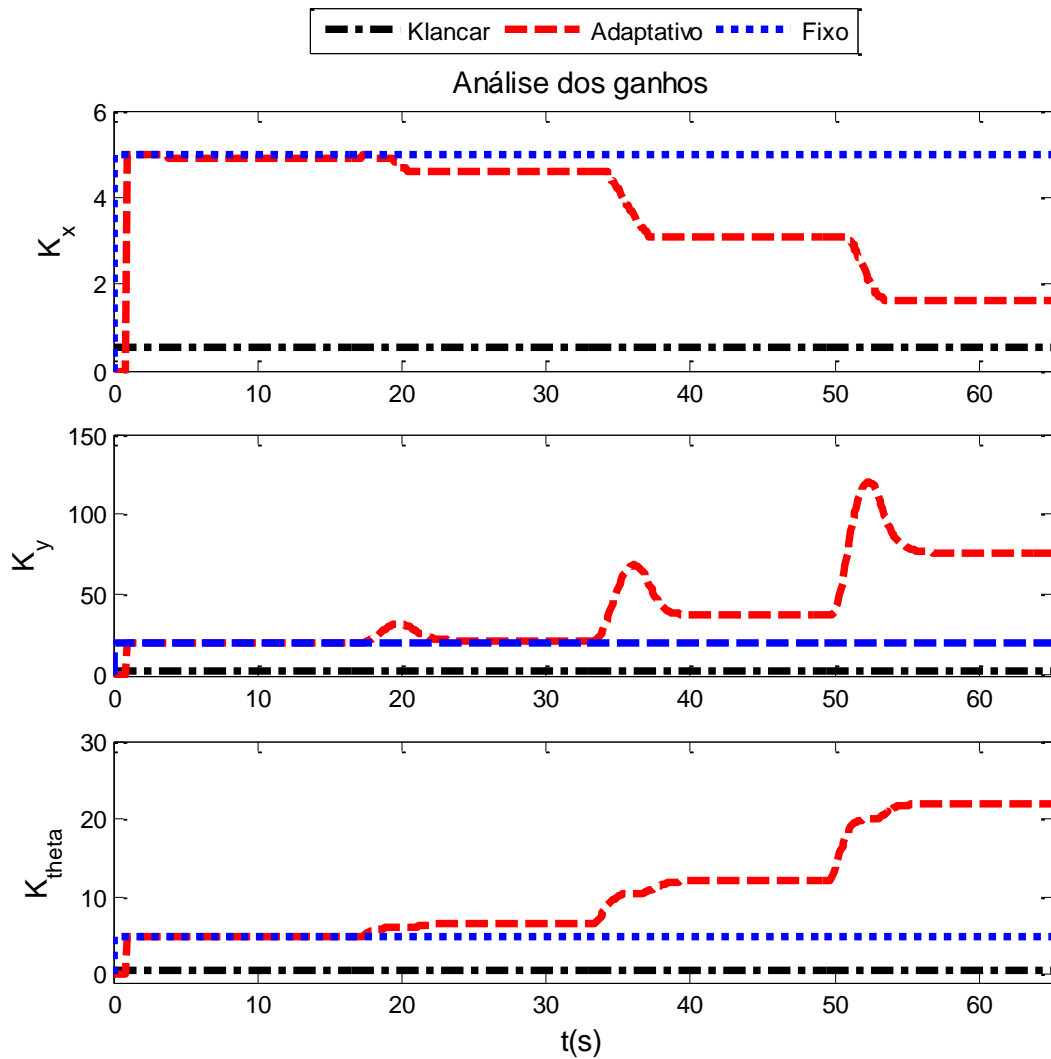
Figura 5-4: Trajetória de referência usada para ajuste dos controladores.



Fonte: Autor.

Os valores dos ganhos dos controladores são apresentados na Figura 5-5. Percebe-se claramente que a redução dos erros controlador adaptativo ao longo da trajetória é devido à adaptação dos ganhos. Por outro lado, como para esta trajetória utiliza valores constantes de velocidade angular e linear, o controlador Klancar apresenta valores constantes de parâmetros.

Figura 5-5: Trajetória de referência usada para ajuste dos controladores.



Fonte: Autor.



## 6 CONCLUSÃO

Neste trabalho, foi apresentado um novo controlador *backstepping* adaptativo com base nas equações exatas da dinâmica dos atuadores e no modelo cinemático de um robô móvel com duas rodas com acionamento diferencial. Analisou-se a forma de como o esquema adaptativo ajusta matematicamente os ganhos do controlador *backstepping online*, minimizando o índice de desempenho de erro de rastreamento para qualquer tipo de trajetória de referência.

As leis de controle analisadas conseguiram alcançar o objetivo de realizar o controle de diferentes trajetórias e em algumas situações adversas, das quais foram consideradas erro de posição inicial e aplicação de distúrbios na dinâmica e na cinemática do sistema. Os resultados simulados e a análise comparativa entre o controlador *backstepping* com ganhos fixos e o controlador *backstepping* adaptativo confirmam a robustez do controlador adaptativo proposto e sua eficácia na melhoria do desempenho de rastreamento do robô móvel.

Conclui-se, com isto, que este trabalho trouxe como contribuição o estudo, projeto e aplicação em um robô móvel não-holonômico com duas rodas de acionamento diferencial de um controlador *backstepping* adaptativo, modelado com base na cinemática e na dinâmica, para problemas de seguimento de diferentes tipos de trajetória. Além de também ter analisado o comportamento deste controlador para problemas relacionados com distúrbios e as incertezas do sistema.

## REFERÊNCIAS

- AMORIM, B. M. M. **Controlador Adaptativo Backstepping a Estrutura Variável com Observador de Estado**. Dissertação (Mestrado) - Universidade Federal do Rio Grande do Norte. Programa de Pós-Graduação em Engenharia Elétrica e de Computação. Natal - RN., p. 100. 2012.
- ANDRADE, F. V. D. **Estudo Comparativo de Controladores Aplicados às Trajetórias de Robôs Móveis**. Dissertação (Mestrado), Universidade Federal do Ceará, Centro de Tecnologia, Departamento de Engenharia Elétrica, Programa de Pós-Graduação em Engenharia Elétrica. Fortaleza. 2011.
- BERTOL, D. W. **Contribuições à Locomoção de Robôs Móveis Não-Holonômicos Usando Controle Fuzzy Baseado em Modelo**. Tese de Doutorado. Programa de Pós-Graduação em Engenharia de Automação e Sistemas - Universidade Federal de Santa Catarina. Florianópolis - SC, p. 148. 2015.
- CRAIG, J. J. **Introduction to Robotics: Mechanics and Control**. [S.l.]: Pearson Education, 2008.
- DAILY Mail. **Daily Mail**, 2016. Disponível em: <<http://www.dailymail.co.uk/sciencetech/article-3595424/US-Navy-buys-blackwing-drones-blasted-air-submarine.html>>. Acesso em: 23 Maio 2016.
- DIAZ, J. F. A. **Estudo e Projeto Conceitual de um Robô para Inspeção de Linhas de Serviço**. UNIVERSIDADE FEDERAL DO RIO DE JANEIRO. RIO DE JANEIRO, p. 97. 2008.
- ENGADGET. **Engadget**, 2016. Disponível em: <<http://www.engadget.com/2016/05/18/saab-underwater-terrorist-bomb-robot/>>. Acesso em: 23 Maio 2016.



FIERRO, R.; LEWIS, F. L. Control of a Nonholonomic Mobile Robot: Backstepping Kinematic Into Dynamics. **Proceedings os the 34th Conference on Decision & Control**, New Orleans, LA, n. IEEE, December 1995.

HUANG, T. J. Adaptive tracking control of high-order non-holonomic mobile robot systems. **IET Control Theory Appl.**, v. 3, p. 681 - 690, 2009.

JÚNIOR, C. D. S.; HERMELY, E. M. Controle de Robôs Móveis Utilizando o Modelo Cinemático. **Revista Controle & Automação.**, v. 14, p. 384 - 392, Outubro 2003.

KANAYAMA, Y. **A Stable Tracking Control Method for an Autonomous Mobile Robot.** Proceedings of the IEEE Conference on Robotics Automation. Santa Bárbara CA, p. 384 - 389. 1990.

KHALIL, H. K. **Nonlinear Systems.** 2. ed. Upper Saddle River: Michigan State Universite, 1996.

KOKOTOVIĆ, P. V. **The Joy of a Feedback: Nonlinear and Adaptive.** Control Systems Magazine, IEEE 12. [S.l.]. 1992.

KUC, T.-Y.; BAEK, S.-M.; PARK, K. Adaptive Learning Controller for Autonomous Moile Robots. **IEE Proc. Control Theory Appl.**, v. 148, p. 49 - 54, January 2001.

MOHARERI, O.; DHAOUADI, R.; RAD, A. B. Indirect adaptive tracking control of a nonholonomic mobile robot via neural networks. **www.elsevier.com/locate/neucom**, 8 Março 2012. 54 - 66.

OGAWA, M. A. **Controle Preditivo Aplicado ao Seguimento de Trajetória de Robô Móvel com Rodas.** Dissertação (Mestrado), Universidade Federal do Ceará, Centro de Tecnologia, Departamento de Engenharia Elétrica, Programa de Pós-Graduação em Engenharia Elétrica. Fortaleza. 2014.

OXFORD Dictionaies. **Oxford University Press**, 2016. Disponível em:

<[http://www.oxforddictionaries.com/definition/american\\_english/robotics](http://www.oxforddictionaries.com/definition/american_english/robotics)>. Acesso em: 30 Maio 2016.

P. COELHO, U. N. Path-following control of mobile robots in presence of uncertainties. **IEEE Trans. Robot.** **21**, p. 252 - 261, 2005.

PORTAL R3. **Portal R3**, 2016. Disponível em:

<<http://www.portalr3.com.br/2016/05/batalhao-em-pinda-recebe-robos-que-serao-usados-nas-olimpiadas/#.V0yNwPkrLDd>>. Acesso em: 23 Maio 2016.

ROMERO, R. A. F. et al. **Robótica Móvel**. 1. ed. Rio de Janeiro: LTC, 2014.

S. DONG, N. H. D. S. K. T. Tracking stabilization of differential mobile robots using adaptive synchronized control. **Proceedings of the IEEE Conference on Robotics and Automation**, Washington, p. 2638 - 2643, May 2002.

SICILIANO, B. et al. **robotic, Modelling, Planning and Control**. Verlag - London: Springer, 2009.

SIEGWART, R.; NOURBAKHSI, I. R. **Introduction to Autonomous Mobile Robots**. 1. ed. London: Brandford Book, 2004.

SOUZA, D. R. **Avaliação estrutural de Dutos com Defeitos de corrosão Reais**. Pontifícia Universidade Católica do Rio de Janeiro PUC. Rio de Janeiro. 2003.

T. DAS, I. N. K. Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots. **IEEE Trans. Control Syst. Technol.** **14**, p. 501 - 510, 2006.

VIEIRA, F. C. **Controle Dinâmico de Robôs Móveis com Acionamento Diferencial**. Universidade Federal do Rio Grande do Norte. Natal, p. 95. 2005.