Roberto Santos Inoue

Controle Robusto de Robôs Móveis com Rodas

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica

Área de Concentração: Sistemas Dinâmicos
Orientador: Prof. Dr. Marco Henrique Terra
Co-orientador: Prof. Dr. Adriano A. G. Siqueira

São Carlos
2007

Sumário

Resumo	vii
Abstract	ix
Publicações	xi
Lista de Figuras	xiii
1 Introdução	1
1.1 Motivação	1
1.2 Revisão bibliográfica	2
1.2.1 Robôs móveis com rodas	2
1.2.2 Controle $\mathcal{H}_\infty$ para robôs	5
1.2.3 Localização usando visão computacional	6
1.3 Disposição dos capítulos	8
2 Modelagem do Robô Móvel	9
2.1 Modelo cinemático	10
2.2 Controlador baseado na cinemática	11
2.3 Modelo dinâmico	12
3 Controle $\mathcal{H}_\infty$ Não Linear	13
3.1 Formulação do problema	13
3.2 Controle $\mathcal{H}_\infty$ não linear via representação quase-LPV	16
3.2.1 Ganho $\mathcal{L}_2$ para sistemas não lineares variantes no tempo	16
3.2.2 Síntese do controle $\mathcal{H}_\infty$ para sistemas LPV por realimentação do estado .	17
3.2.3 Considerações computacionais	18
3.3 Controle $\mathcal{H}_\infty$ não linear via Teoria dos Jogos (TJ)	20
3.4 Controle $\mathcal{H}_\infty$ não linear baseado em modelo fuzzy Takagi Sugeno	22

3.5	Controle $\mathcal{H}_\infty$ não linear baseado em redes neurais	30
3.6	Solução da equação algébrica de Riccati	34
4	Implementação	37
4.1	Robô móvel com rodas	37
4.2	Sistema de visão computacional	38
4.2.1	Transformada de Hough circular	39
4.3	Ambiente de controle de RMR	41
4.4	Experimentos	43
4.4.1	Controle $\mathcal{H}_\infty$ não linear via representação quase-LPV	46
4.4.2	Controle $\mathcal{H}_\infty$ não linear via Teoria dos Jogos (TJ)	50
4.4.3	Controle $\mathcal{H}_\infty$ não linear baseado em modelo fuzzy Takagi Sugeno	53
4.4.4	Controle $\mathcal{H}_\infty$ não linear baseado em modelo e em modelo fuzzy Takagi Sugeno	57
4.4.5	Controle $\mathcal{H}_\infty$ não linear baseado em redes neurais	61
4.4.6	Controle $\mathcal{H}_\infty$ não linear baseado em modelo e em redes neurais	65
4.4.7	Controle proporcional derivativo e torque calculado (PD+TC)	69
4.4.8	Estudo comparativo	72
5	Conclusão	75
	Referências Bibliográficas	77

Aos meus pais Francisco Hitoshi Inoue e Maria Selma dos Santos com amor e gratidão.

A Deus, que me concedeu saúde física e mental para a realização deste trabalho.

Aos meus pais Francisco Hitoshi Inoue e Maria Selma dos Santos com amor, admiração e gratidão por sua compreensão, carinho, presença e incansável apoio ao longo do período deste mestrado.

Aos meus tios Gilberto e Djalma que me ajudaram em minha permanência em São Carlos.

Ao Prof Dr. Marco Henrique Terra pela confiança, orientação, paciência e pelo tempo dedicado a este trabalho.

Ao Prof. Dr. Adriano Almeida Gonçalves Siqueira pela atenção, apoio e contribuições na realização deste trabalho.

Aos amigos da pós-graduação e do LASI: Aline, Amanda, Carolina, Cleber, Elmer, Gildson, Lais, Rafael, Raphael, Robson, Samuel, Saulo, Tatiana, Tatiane, Thiago, Wallisson, pela amizade, paciência, companheirismo e colaborações durante a realização das disciplinas e deste trabalho.

Aos professores e funcionários do Departamento de Engenharia Elétrica da Escola de Engenharia de São Carlos, pelas contribuições durante o mestrado.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela concessão da bolsa de mestrado.

Resumo

Nesta dissertação é apresentado um estudo comparativo entre seis controladores $\mathcal{H}_\infty$ não lineares aplicados em um robô móvel com rodas. Três estratégias de controle são avaliadas. Na primeira, o modelo do robô é considerado completamente conhecido. Na segunda, o modelo matemático é considerado desconhecido e é realizada uma estimativa baseada em métodos inteligentes. E finalmente, na terceira estratégia, o modelo nominal é conhecido e técnicas inteligentes são usadas para estimar somente incertezas paramétricas do robô. As técnicas inteligentes usadas são baseadas em redes neurais e em lógica fuzzy. Esses controladores são resolvidos através de Desigualdades Matriciais Lineares (DMLs) e equações algébricas de Riccati. Todos os resultados obtidos são baseados em dados experimentais.

Abstract

This dissertation is present a comparative study between six nonlinear $\mathcal{H}_\infty$ controllers applied to a wheeled mobile robot. Three control strategies are adopted. In the first, the model of the robot is considered completely known. In the second, the mathematical model is considered unknown and is accomplished an estimate based on intelligent methods. And finally, in the third strategy, the nominal model is known and intelligent techniques are used only to estimate parametric uncertainties of the robot. The intelligent techniques used are based in neural networks and in fuzzy logic. These controllers are solved via Linear Matrix Inequalities (LMIs) and algebraic Riccati equations. All results obtained are based in experimental data.

Publicações

1. Inoue, R. S., A. A. G. Siqueira e M. H. Terra (2007). Experimental results on the nonlinear $\mathcal{H}_\infty$ control via Quasi-LPV representation and game theory for wheeled mobile robots. *IEEE Conference on Control Applications*, Singapore.
2. Inoue, R. S., Tatiana de F. P. A. T. Pazelli, A. A. G. Siqueira e M. H. Terra (2007). Mixed model based/fuzzy adaptive robust controller with $\mathcal{H}_\infty$ criterion applied to wheeled mobile. *3rd IFAC Symposium on System Structure and Control*, Foz do Iguaçu, Brazil.
3. Inoue, R. S., Tatiana de F. P. A. T. Pazelli, A. A. G. Siqueira e M. H. Terra (2007). Controlador Robusto $\mathcal{H}_\infty$ Baseado em Redes Neurais Aplicado em Robôs Móveis com Rodas. *VIII Simpósio Brasileiro de Automação Inteligente*, Florianópolis, Brasil.
4. Pazelli, Tatiana de F. P. A. T., R. S. Inoue, A. A. G. Siqueira e M. H. Terra (2007). Mixed Model Based/Fuzzy Adaptive Robust Controller with $\mathcal{H}_\infty$ Criterion Applied to Free-Floating Space Manipulators. *VIII Simpósio Brasileiro de Automação Inteligente*, Florianópolis, Brasil.

Lista de Figuras

FIGURA 2.1	Geometria do RMR.	9
FIGURA 3.1	Rede Neural.	32
FIGURA 4.1	Foto do robô móvel.	38
FIGURA 4.2	Convenção: (a) imagem contínua, (b) imagem digital.	39
FIGURA 4.3	Transformada de Hough circular: (a) imagem, (b) matriz acumuladora.	39
FIGURA 4.4	Imagem da câmera visualizando o robô.	41
FIGURA 4.5	Ambiente de controle de RMR na aba de Parâmetros.	42
FIGURA 4.6	Ambiente de controle de RMR na aba de Gráfico A.	42
FIGURA 4.7	Ambiente de controle de RMR na aba de Gráfico B.	43
FIGURA 4.8	Distúrbios de torque aplicados às rodas.	44
FIGURA 4.9	Sistema de controle de acompanhamento de trajetória para robôs móveis.	45
FIGURA 4.10	Controladores baseados na dinâmica	45
FIGURA 4.11	Acompanhamento de trajetória de referência usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	47
FIGURA 4.12	Erros de posição usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	47
FIGURA 4.13	Erro de direção usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	48

FIGURA 4.14 Derivada dos erros de posição usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	48
FIGURA 4.15 Derivada do erro de direção usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	48
FIGURA 4.16 Velocidade angular da roda direita usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	49
FIGURA 4.17 Velocidade angular da roda esquerda usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	49
FIGURA 4.18 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).	49
FIGURA 4.19 Acompanhamento de trajetória de referência usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	50
FIGURA 4.20 Erros de posição usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	50
FIGURA 4.21 Erro de direção usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	51
FIGURA 4.22 Derivada dos erros de posição usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	51
FIGURA 4.23 Derivada do erro de direção usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	51
FIGURA 4.24 Velocidade angular da roda direita usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	52
FIGURA 4.25 Velocidade angular da roda esquerda usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	52
FIGURA 4.26 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).	52
FIGURA 4.27 Conjunto fuzzy $A_1(\tilde{q})$ e conjunto fuzzy $A_2(\dot{\tilde{q}})$	53
FIGURA 4.28 Acompanhamento de trajetória de referência usando o controlador ba- seado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	54

FIGURA 4.29 Erros de posição usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	54
FIGURA 4.30 Erro de direção usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	55
FIGURA 4.31 Derivada dos erros de posição usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	55
FIGURA 4.32 Derivada do erro de direção usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	55
FIGURA 4.33 Velocidades angulares da roda direita usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	56
FIGURA 4.34 Velocidades angulares da roda esquerda usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	56
FIGURA 4.35 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	56
FIGURA 4.36 Conjunto fuzzy $A_1(\tilde{q})$ e conjunto fuzzy $A_2(\dot{\tilde{q}})$	57
FIGURA 4.37 Acompanhamento de trajetória de referência usando o controlador base- ado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	58
FIGURA 4.38 Erros de posição usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	58
FIGURA 4.39 Erro de direção usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	59
FIGURA 4.40 Derivada dos erros de posição usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	59
FIGURA 4.41 Derivada do erro de direção usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	59
FIGURA 4.42 Velocidades angulares da roda direita usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	60

FIGURA 4.43 Velocidades angulares da roda esquerda usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	60
FIGURA 4.44 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).	60
FIGURA 4.45 Acompanhamento de trajetória de referência usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita). . . .	62
FIGURA 4.46 Erros de posição usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	62
FIGURA 4.47 Erro de direção usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	62
FIGURA 4.48 Derivada dos erros de posição usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	63
FIGURA 4.49 Derivada do erro de direção usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	63
FIGURA 4.50 Velocidades angulares da roda direita usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	63
FIGURA 4.51 Velocidades angulares da roda esquerda usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	64
FIGURA 4.52 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	64
FIGURA 4.53 Acompanhamento de trajetória de referência usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	66
FIGURA 4.54 Erros de posição usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	66
FIGURA 4.55 Erro de direção usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	66

FIGURA 4.56 Derivada dos erros de posição usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	67
FIGURA 4.57 Derivada do erro de direção usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).	67
FIGURA 4.58 Velocidades angulares da roda direita usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita). . .	67
FIGURA 4.59 Velocidades angulares da roda esquerda usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita). .	68
FIGURA 4.60 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo e em redes neurais: sem distúrbio (es- querda) e com distúrbio (direita).	68
FIGURA 4.61 Acompanhamento de trajetória de referência usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).	69
FIGURA 4.62 Erros de posição usando o controlador PD + TC: sem distúrbio (es- querda) e com distúrbio (direita).	69
FIGURA 4.63 Erro de direção usando o controlador PD + TC: sem distúrbio (es- querda) e com distúrbio (direita).	70
FIGURA 4.64 Derivada dos erros de posição usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).	70
FIGURA 4.65 Derivada do erro de direção usando o controlador PD: sem distúrbio (esquerda) e com distúrbio (direita).	70
FIGURA 4.66 Velocidades angulares da roda direita usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).	71
FIGURA 4.67 Velocidades angulares da roda esquerda usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).	71
FIGURA 4.68 Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita). .	71

Capítulo 1

Introdução

1.1 Motivação

Considerando que robôs móveis com rodas (RMRs) podem estar sujeitos a perturbações externas, como desníveis da superfície por onde o robô circula, escorregamento das rodas, colisão com obstáculos ou com outros robôs e incertezas paramétricas, o controle utilizado deve ser robusto o suficiente para a realização de uma tarefa pré-estabelecida. Optou-se neste trabalho pela utilização de estratégias de controle baseadas em modelos dinâmicos. Será usada uma representação do robô em espaço de estado linear a parâmetros variantes (LPV). Essa representação tem se mostrado bastante apropriada em aplicações voltadas à robótica pois preserva a natureza dinâmica do sistema. Para a atenuação dos distúrbios externos e incertezas paramétricas, o critério $\mathcal{H}_\infty$ tem sido usado para outros tipos de robôs e será aplicado em RMRs neste trabalho. Em adição a este caráter robusto do critério de controle adotado, este trabalho utiliza métodos baseados em inteligência artificial, baseados em redes neurais e em lógica fuzzy, para complementar o modelo matemático do robô móvel. Estimam as incertezas não modeladas pelo modelo nominal. Esta combinação é feita com a garantia de estabilidade do sistema de controle robusto resultante.

1.2 Revisão bibliográfica

1.2.1 Robôs móveis com rodas

Controladores para RMRs têm sido alvo de pesquisas em robótica a partir dos anos 80. A expressão Robôs Móveis com Rodas será utilizada para diferenciar a categoria de robôs considerada neste trabalho de outros tipos de robôs (móveis, aquáticos, aéreos, etc) e em boa parte do texto será utilizado apenas o termo Robôs Móveis.

Em *Campion et al. (1996)*, os autores definiram dois tipos de rodas para RMRs: *rodas convencionais*, cuja velocidade no ponto de contato da roda com o solo é zero e são divididas em rodas fixas, centradas orientáveis e centradas não orientáveis (conhecidas também por castor); e *rodas suecas* nas quais somente a componente da velocidade ao longo do movimento no ponto de contato da roda com o solo é admitida ser nula.

Os RMRs mais comuns estudados na literatura são: unicycle (*Aicardi et al., 1995; Morin e Samson, 2000; Lee et al., 2001*) (este nome é devido a equação cinemática do robô ser equivalente ao de uma roda que não gira em falso e nem desliza no sentido do eixo), carro convencional (*Almeida et al., 1997*), carro convencional com *trailers* (*Vendittelli e Oriolo, 2000; Samson, 1995; Jiang e Nijmeijer, 1999*) e unicycle com *trailers* (*M'Closkey e Murray, 1997*).

O robô móvel utilizado neste trabalho é um unicycle com duas rodas convencionais fixas atuadas independentemente e uma roda convencional tipo castor. Considera-se também que o centro de massa (P_c) é diferente do ponto no centro do eixo das rodas atuadas (P_o).

Modelos matemáticos para RMRs têm sido formulados para cada tipo de robô. Modelos cinemáticos para cinco tipos de robôs são apresentados em *Campion et al. (1996)*. Em *Coelho (2001)*, o autor apresenta uma modelagem completa (cinemática e dinâmica) para um robô unicycle, considerando três casos: no primeiro $P_c = P_o$, no segundo $P_c \neq P_o$ e no terceiro $P_c \neq P_o$, incluindo em todos eles as forças de restrições.

Robôs Móveis com Rodas constituem uma classe de sistemas mecânicos caracterizados por restrições não holonômicas. Para entender o que são restrições não holonômicas, considere que o sistema mecânico pode ser descrito por um vetor de coordenadas generalizadas de dimensão n

$$q = [q_1 \ q_2 \ \dots \ q_n]^T,$$

e a velocidade generalizada em um ponto genérico de uma trajetória suave $q(t)$ é o vetor tangente

$$\dot{q} = [\dot{q}_1 \quad \dot{q}_2 \quad \dots \quad \dot{q}_n]^T.$$

Um sistema mecânico pode estar sujeito a um conjunto de restrições cinemáticas, envolvendo as coordenadas generalizadas e suas derivadas, ou seja, k restrições de primeira ordem

$$\phi_i^T(q, \dot{q}) = 0,$$

sendo $i = \{1, 2, \dots, k\}$. Na maioria dos casos as restrições são lineares nas velocidades e podem ser denotadas por

$$A(q)\dot{q} = 0.$$

Se as restrições cinemáticas podem ser integradas, isto é, se existem k funções h_i tais que

$$\frac{dh_i(q(t))}{dt} = \frac{\partial h_i(q(t))}{\partial q} \dot{q} = 0, \quad i = \{1, 2, \dots, k\},$$

então, as restrições cinemáticas são de fato restrições geométricas. Portanto um conjunto de restrições cinemáticas é chamado holonômico se é integrável e em caso contrário, é chamado de não holonômico.

Usando Álgebra de Lie, Coelho e Nunes (2003) mostraram como pode-se determinar quantas restrições cinemáticas não holonômicas estão presentes em um sistema de restrições. Uma limitação na estabilidade de um ponto de equilíbrio de um sistema com restrições não holonômicas é que a estabilidade de Lyapunov não pode ser alcançada por uma lei de realimentação invariante no tempo e suave (Bloch e McClamroch, 1989; Oriolo et al., 2002). Este resultado é baseado nos resultados apresentados em Brockett (1983).

Estudos de controlabilidade e estabilizabilidade para sistemas não holonômicos podem ser encontrados em Bloch e McClamroch (1990), onde se utiliza como exemplo o movimento de uma faca, e em Oriolo et al. (2002), que avalia um robô do tipo unicycle.

Na literatura são apresentados três tipos básicos de objetivos a serem alcançados por um RMR: *estabilização de postura* (posição e direção do robô), alcançar uma postura de referência iniciando em uma dada postura; *acompanhamento de trajetória*, o robô deve seguir uma trajetória de referência em função do tempo; *seguindo um caminho*, o robô deve seguir uma tra-

jetória de referência em função de parâmetros independentes do tempo, podendo ser geométrico (Coelho e Nunes, 2003) ou em função do trajeto e das velocidades ao longo do caminho (Sarkar et al., 1994). Neste trabalho, o controle será realizado para acompanhamento de trajetória de referência, sendo a trajetória de referência o objetivo a ser alcançado pelo ponto P_c do robô, diferentemente da trajetória desejada, que se refere às velocidades desejadas para as rodas do robô tais que o robô alcance a referência, conforme apresentado no Capítulo 2.2.

Na literatura foram encontradas inúmeras publicações de trabalhos relacionados a controle de RMRs e alguns estão resumidos a seguir.

Oriolo et al. (2002) apresentaram um método unificado para resolver os problemas de acompanhamento de trajetória e estabilização de postura, que utiliza o modelo cinemático de um uniclo e a linearização por realimentação dinâmica. O método consiste em encontrar um compensador dinâmico dependente da direção do robô e das variáveis de controle geradas por um controlador proporcional derivativo (PD).

Sarkar et al. (1994) propuseram um método de controle por realimentação de saída que alcança estabilidade assintótica usando o modelo cinemático e dinâmico. A escolha das saídas está ligada diretamente ao objetivo de controle. Para o acompanhamento de trajetória, o vetor de saída consiste de parte das coordenadas generalizadas em função do tempo, e para seguir um caminho, o vetor de saída consiste em um caminho geométrico, por exemplo uma reta ou uma circunferência.

Coelho e Nunes (2003) mostraram o uso da álgebra de Lie para sistemas de controle não lineares com restrições não holonômicas. O controle de um RMR é realizado por realimentação de saída, sendo as saídas o erro de posição do robô com relação a uma circunferência e a velocidade linear do robô para o problema de seguir um caminho.

Fukao et al. (2000) apresentaram um método para projetar um controlador adaptativo para o modelo dinâmico de um robô móvel com restrições não holonômicas com parâmetros desconhecidos, utilizando *Backstepping* adaptativo. A metodologia de projeto recursiva chamada *Backstepping*, (Krstic et al., 1995), é uma construção sistemática de ambas as leis de controle de realimentação e função de Lyapunov associada. Propriedades fortes de acompanhamento e estabilidade local ou global são construídas dentro de sistemas não lineares em um número de passos que nunca é maior do que a ordem do sistema.

Do et al. (2004) apresentaram um controle adaptativo global variante no tempo que resolve simultaneamente ambas estabilização de postura e acompanhamento de trajetória para robôs

móveis com parâmetros da dinâmica e cinemática desconhecidos. A síntese de controle é baseada em técnicas de *Backstepping* e método direto de Lyapunov.

Ji et al. (2003) desenvolveram um sistema de controle adaptativo tolerante a falha. O robô é modelado como um sistema contínuo com um controlador supervisor. Um ajuste em um controle hierárquico é desenvolvido para o controlador supervisor que determina uma estratégia de controle adequada para eliminar a falha.

Através de uma função de Lyapunov, Kanayama et al. (1990) propuseram um controle de acompanhamento de trajetória estável, que envolve o erro de postura (erro de direção e posição com relação ao sistema de coordenadas local do robô) e a trajetória de referência, e gera velocidades desejadas para o robô móvel seguir a trajetória de referência, ou seja, zerar o erro de postura.

1.2.2 Controle $\mathcal{H}_\infty$ para robôs

Três importantes estratégias sobre controle de sistemas robóticos têm sido tratadas na literatura: na primeira, o modelo de um sistema robótico é considerado completamente conhecido e utilizável para o controlador (Lewis et al., 1993); na segunda, os parâmetros do modelo são desconhecidos e eles são estimados baseados na propriedade robótica de parametrização linear, resultados clássicos sobre controle adaptativo podem ser vistos em Craig (1985) e Lewis et al. (1993); e na terceira, o modelo é desconhecido e uma abordagem inteligente (baseada em redes neurais ou lógica fuzzy) é usada para estimar o modelo (veja, por exemplo, Chang (2000) e Chang (2005)). Se, em adição à incertezas paramétricas, distúrbios externos estão presentes, a dificuldade para controlar um sistema robótico sob estas circunstâncias aumentam. Uma abordagem interessante para resolver este problema de controle é baseado no critério $\mathcal{H}_\infty$ não linear, que visa atenuar os efeitos de todos distúrbios no desempenho do sistema.

Controladores $\mathcal{H}_\infty$ baseados nas estratégias descritas acima são propostos em Chen et al. (1994); Wu (1995); Wu et al. (1996); Chen et al. (1997); Chang e Chen (1997); Huang e Jadbabaie (1998); Chang (2000); Siqueira e Terra (2004); Chang (2005), para robôs manipuladores. Em Chen et al. (1994), uma solução explícita para o problema de controle $\mathcal{H}_\infty$ não linear, onde o modelo do manipulador é considerado completamente conhecido, é desenvolvida baseada na Teoria dos Jogos (TJ) (em Postlethwaite e Bartoszewicz (1998) uma metodologia similar é usada para controlar um manipulador real). Controladores $\mathcal{H}_\infty$ não lineares para sistemas LPVs, que têm sido aplicados em robôs manipuladores (veja por exemplo Siqueira e Terra (2004)), podem

ser vistos em Wu (1995); Wu et al. (1996); Huang e Jadbabaie (1998).

Um algoritmo de controle adaptativo $\mathcal{H}_\infty$ não linear é proposto em Chen et al. (1997), onde um projeto de controle robusto de acompanhamento de trajetória considera que os parâmetros desconhecidos podem ser aprendidos por uma lei clássica de adaptação atualizável. Controles adaptativos $\mathcal{H}_\infty$ não lineares baseados em técnicas inteligentes podem ser vistos em Chang e Chen (1997), Chang (2000) e Chang (2005). Redes neurais e lógica fuzzy são empregadas para estimar todo o modelo dinâmico do robô. Uma abordagem interessante desenvolvida em Ge et al. (1998) usa um controlador adaptativo baseado no modelo nominal e em redes neurais, sendo que as redes neurais são usadas somente para estimar as incertezas paramétricas do sistema robótico. Vale destacar que este controlador não utiliza um critério de desempenho robusto, semelhante ao critério $\mathcal{H}_\infty$ por exemplo.

Poucos trabalhos têm sido apresentados com técnicas de controle $\mathcal{H}_\infty$ direcionados a RMRs. Hwang et al. (2004) propuseram uma combinação de um controlador baseado na cinemática e um controlador $\mathcal{H}_\infty$ robusto baseado na dinâmica para acompanhamento de trajetória. A solução proposta em Hwang et al. (2004) resulta em matrizes constantes do controlador, não leva em consideração a natureza variante dos parâmetros do robô. Já em dos Reis (2005), equações dinâmicas de um RMR são descritas na forma quase-LPV. Um controlador baseado no modelo cinemático, proposto em Kanayama et al. (1990), é utilizado para gerar as velocidades desejadas para as rodas e controladores robustos baseados no modelo dinâmico são projetados levando em consideração as variações paramétricas do robô.

Neste trabalho, que complementa os resultados obtidos em dos Reis (2005); Ge et al. (1998); Chang (2005), é feito um estudo comparativo, utilizando resultados experimentais de um RMR, entre dois controladores $\mathcal{H}_\infty$ apresentados em dos Reis (2005), quatro controladores robustos baseados em técnicas inteligentes e um controlador Proporcional Derivativo mais torque calculado. Os controladores robustos baseados em técnicas inteligentes, utilizam redes neurais e o modelo fuzzy Takagi Sugeno (T-S) (Takagi e Sugeno, 1985) para estimar todo o modelo nominal do RMR ou para aprender apenas as incertezas do modelo nominal.

1.2.3 Localização usando visão computacional

A determinação da postura de RMRs é uma área de pesquisa de destaque para a implementação dos controladores, pois uma boa precisão da posição e direção é de fundamental importância para o funcionamento do sistema.

Os sensores mais utilizados são os *encoders* (por exemplo Oriolo et al. (2002); Sarkar et al. (1994)), sonares (Bozma e Kuc, 1994) ou a união destes dois (Corradini et al., 2003). A utilização de câmeras está sendo largamente pesquisada, sendo que elas podem ser embarcadas (Lee et al., 2004), como em visão omnidirecional que produzem imagens de 360° do ambiente (Yagi et al., 1994), ou externas como em Gupta et al. (2005); Liu et al. (2004), na qual a câmera fica em cima de um campo de atuação (visão global).

Posições em imagens 2D podem ser obtidas com extração de características como pontos, linhas, áreas ou formas geométricas. Um procedimento para detecção aproximada de círculos e arcos em imagens monocromática (escala em níveis de cinza) foi mostrado em Kimme e Sklansky (1975), fazendo primeiro uma extração de bordas e depois aplicando a Transformada de Hough com matrizes tridimensionais de acumuladores. Ho e Chen (1995) utilizaram simetria geométrica para encontrar círculos/elipses em imagens após aplicar um processo de extração de bordas. Um método para detecção de características geométricas (linhas, círculos e elipses) foi proposto por McLaughlin e Alder (1998) e comparado com a transformada de Hough.

Para um RMR seguir um caminho desenhado na superfície, Bianchi et al. (2001) utilizaram uma câmera de *vídeo* colorida sobre a área de movimentação do robô (visão global). A identificação da posição, direção e caminho a seguir são encontrados particionando a imagem em regiões de cores de *pixels* usando a técnica de imposições de limiares.

Luca et al. (2002) utilizaram a informação visual de uma câmera no teto para identificar a postura de um RMR. Em cima do robô foi criada uma superfície de cor preta com três *LEDs* localizados nos vértices de um triângulo isósceles. O método consiste primeiro em tornar binária a imagem monocromática capturada, depois aplica-se um operador dilatação (Sonka et al., 1998) e extração das regiões de luminosidades dos *LEDs* e por último as posições dos vértices do triângulo são encontradas por um algoritmo baseado na relação das distâncias entre as posições candidatas a vértices. Portanto, a posição será um dos vértices e a direção dada por relações trigonométricas.

Este trabalho utiliza para a localização de um RMR real o método da Transformada de Hough Circular (THC), mostrado em Sonka et al. (1998). O método é baseado na determinação de centros de círculos de imagens monocromáticas obtidas da parte superior do RMR. A determinação da postura está apresentada no Capítulo 4.2.

1.3 Disposição dos capítulos

No **Capítulo 2** é descrita a modelagem completa, cinemática e dinâmica, do robô móvel, o controle cinemático proposto por Kanayama et al. (1990) para obter as velocidades angulares desejadas, dadas a postura atual e a trajetória de referência.

No **Capítulo 3** é abordado o projeto de dois controladores robustos baseados no modelo dinâmico usando técnicas de controle $\mathcal{H}_\infty$ para RMRs e quatro controladores robustos usando técnicas de controle $\mathcal{H}_\infty$, redes neurais e modelo fuzzy T-S.

No **Capítulo 4** são mostrados o RMR, o sistema de visão computacional, o ambiente de controle de RMR e os resultados experimentais do controle do RMR.

No **Capítulo 5** são apresentadas as conclusões.

angulares das rodas direita e esquerda, respectivamente.

A modelagem do robô apresentada a seguir está baseada em Coelho (2001) e dos Reis (2005).

2.1 Modelo cinemático

O RMR mostrado na Fig. 2.1 apresenta três restrições cinemáticas (Coelho e Nunes, 2003). Assume-se que o robô não pode deslizar, ou seja, movimentar-se na direção do eixo das rodas atuadas. A velocidade de P_o deve estar na direção do eixo de simetria (eixo X_o), assim, a primeira restrição com respeito a P_c é dada por

$$\dot{y}_c \cos \alpha - \dot{x}_c \sin \alpha - d\dot{\alpha} = 0,$$

sendo (x_c, y_c) as coordenadas do centro de massa P_c no sistema de coordenadas inercial, e α o ângulo entre o eixo de simetria do robô e o eixo X. As outras duas restrições estão relacionadas com a rotação das rodas, ou seja, as rodas atuadas não podem girar em falso,

$$\dot{x}_c \cos \alpha + \dot{y}_c \sin \alpha + b\dot{\alpha} - r\dot{\theta}_d = 0,$$

$$\dot{x}_c \cos \alpha + \dot{y}_c \sin \alpha - b\dot{\alpha} - r\dot{\theta}_e = 0.$$

Definindo $q = [q_1 \ q_2]$, $q_1 = [x_c \ y_c \ \alpha]^T$ e $q_2 = [\theta_d \ \theta_e]^T$, então as três restrições podem ser escritas na forma

$$A(q)\dot{q} = \begin{bmatrix} -\sin \alpha & \cos \alpha & -d & 0 & 0 \\ -\cos \alpha & -\sin \alpha & -b & r & 0 \\ -\cos \alpha & -\sin \alpha & b & 0 & r \end{bmatrix} \dot{q} = 0.$$

A matriz $A(q)$ tem posto completo e pode ser expressa como $[A_1(q)_{(3 \times 3)} \ A_2(q)_{(3 \times 2)}]$, tal que $A_1(q)$ é não singular e $S(q) = [-A_1^{-1}(q)A_2 \quad I_{(2 \times 2)}]^T$ cujas colunas são o espaço nulo de $A(q)$, ou seja, $A(q)S(q) = 0$. Então encontra-se

$$S(q) = \begin{bmatrix} c(b \cos \alpha - d \sin \alpha) & c(b \cos \alpha + d \sin \alpha) \\ c(b \sin \alpha + d \cos \alpha) & c(b \sin \alpha - d \cos \alpha) \\ c & -c \\ 1 & 0 \\ 0 & 1 \end{bmatrix},$$

sendo $c = r/(2b)$. A equação cinemática é dada por

$$\dot{q}(t) = S(q)\dot{q}_2(t) \quad (2.1)$$

ou

$$\dot{x}_c = c(b \cos \alpha - d \sin \alpha)\dot{\theta}_d + c(b \cos \alpha + d \sin \alpha)\dot{\theta}_e, \quad (2.2)$$

$$\dot{y}_c = c(b \sin \alpha + d \cos \alpha)\dot{\theta}_d + c(b \sin \alpha - d \cos \alpha)\dot{\theta}_e, \quad (2.3)$$

$$\dot{\alpha} = c(\dot{\theta}_d - \dot{\theta}_e). \quad (2.4)$$

2.2 Controlador baseado na cinemática

Nesta seção, é apresentado o controlador baseado na cinemática proposto por Kanayama et al. (1990), que gera as velocidades desejadas para o problema de acompanhamento de trajetória de referência de RMRs. Considere o erro $q_e = [x_e \ y_e \ \alpha_e]^T$, entre a postura de referência $P_r = [x_r, y_r, \alpha_r]^T$, e a postura real do RMR $P_c = [x_c, y_c, \alpha]^T$, dada por

$$\begin{aligned} x_e &= \cos \alpha (x_r - x_c) + \sin \alpha (y_r - y_c), \\ y_e &= -\sin \alpha (x_r - x_c) + \cos \alpha (y_r - y_c), \\ \alpha_e &= \alpha_r - \alpha, \end{aligned} \quad (2.5)$$

sendo $[x_r, y_r]^T = q_r$ a trajetória de referência escolhida e $\alpha_r = \tan^{-1}(\dot{y}_r/\dot{x}_r)$. As velocidades desejadas linear (v^d) e angular (w^d) de um RMR são dadas por

$$\begin{aligned} v^d &= v_r \cos \alpha_e + k_x x_e, \\ w^d &= w_r + v_r(k_y y_e + k_\alpha \sin \alpha_e), \end{aligned} \quad (2.6)$$

sendo k_x, k_y, k_α constantes,

$$v_r = \sqrt{(\dot{x}_r)^2 + (\dot{y}_r)^2} \text{ e } w_r = \dot{\alpha}_r. \quad (2.7)$$

No Capítulo 3, os controladores baseados no modelo dinâmico do robô são resolvidos considerando as velocidades angulares desejadas das rodas, $\dot{q}_2^d$. Então, é necessário definir a relação de velocidades que segue

$$\dot{q}_2^d = \begin{bmatrix} \dot{\theta}_d^d \\ \dot{\theta}_e^d \end{bmatrix} = \begin{bmatrix} 1/r & b/r \\ 1/r & -b/r \end{bmatrix} \begin{bmatrix} v^d \\ w^d \end{bmatrix}, \quad (2.8)$$

sendo $\dot{\theta}_d^d$ e $\dot{\theta}_e^d$ as velocidades angulares desejadas das rodas direita e esquerda, respectivamente.

2.3 Modelo dinâmico

As equações dinâmicas de RMRs usando a teoria de Lagrange é descrita por Coelho e Nunes (2003) como

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = E\tau - A^T(q)\lambda, \quad (2.9)$$

sendo $\lambda = [\lambda_1 \ \lambda_2 \ \lambda_3]^T$ o vetor de restrições de força; $E = [0_{(2 \times 3)} \ I_{(2 \times 2)}]^T$ a matriz da entrada; $\tau = [\tau_d \ \tau_e]^T$ o vetor de entrada (torque nas rodas);

$$C(q, \dot{q}) = \begin{bmatrix} 0 & 0 & m d \dot{\alpha} \cos \alpha & 0 & 0 \\ 0 & 0 & m d \dot{\alpha} \sin \alpha & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

a matriz de forças de Coriolis e centrípeta; e

$$M(q) = \begin{bmatrix} m & 0 & m d \sin \alpha & 0 & 0 \\ 0 & m & -m d \cos \alpha & 0 & 0 \\ m d \sin \alpha & -m d \cos \alpha & I & 0 & 0 \\ 0 & 0 & 0 & I_r & 0 \\ 0 & 0 & 0 & 0 & I_r \end{bmatrix}$$

a matriz de inércia. Os parâmetros m e I são dados por

$$m = m_p + 2m_r \text{ e } I = I_c + 2m_r(d^2 + b^2) + 2I_m + m_p d^2,$$

sendo m_r a massa de cada roda atuada mais o rotor do motor; m_p a massa da plataforma do robô; I_c o momento de inércia da plataforma do robô em relação ao eixo vertical em P_c ; I_r o momento de inércia de cada roda com o rotor em relação ao eixo da roda e I_m o momento de inércia em relação ao eixo definido no plano da roda (perpendicular ao eixo da roda).

Capítulo 3

Controle $\mathcal{H}_\infty$ Não Linear

Uma vez encontradas as velocidades desejadas do robô (v^d e ω^d) utilizando o controlador cinemático e posteriormente as velocidades angulares desejadas das rodas ($\dot{\theta}_d^d$ e $\dot{\theta}_e^d$), torna-se possível aplicar técnicas de controle $\mathcal{H}_\infty$ ao modelo dinâmico para obter os torques a serem aplicados às rodas para o acompanhamento da trajetória de referência. Neste capítulo, serão apresentados os dois controladores $\mathcal{H}_\infty$ não lineares mostrados em dos Reis (2005). Dois controladores robustos baseados em redes neurais e em modelo fuzzy Takagi Sugeno (T-S) que estimam todo o modelo nominal do RMR. E duas abordagens utilizando técnicas de controle baseadas em modelo nominal do RMR e em inteligência artificial. Neste caso, redes neurais e lógica fuzzy são utilizadas para estimar somente as incertezas do modelo nominal.

3.1 Formulação do problema

O modelo dinâmico mostrado na Seção 2.3 é descrito em espaço de estado na forma quase linear a parâmetros variantes (quase-LPV), ou seja, os parâmetros das matrizes do sistema dependem dos estados. Diferenciando (2.1) com respeito ao tempo, tem-se

$$\ddot{q} = \dot{S}(q)\dot{q}_2 + S(q)\ddot{q}_2. \quad (3.1)$$

Substituindo (3.1) e (2.1) na equação (2.9) e multiplicando por S^T à esquerda, obtém-se

$$M_2\ddot{q}_2 + C_2(\dot{q}_2)\dot{q}_2 = S^T E \tau = \tau. \quad (3.2)$$

Incertezas paramétricas são introduzidas em (3.2) dividindo as matrizes de parâmetros M_2 e $C_2(\dot{q}_2)$ em uma parte nominal e uma parte perturbada

$$\begin{aligned} M_2 &= M_0 + \Delta M_0 \\ C_2(\dot{q}_2) &= C_0(\dot{q}_2) + \Delta C_0(\dot{q}_2) \end{aligned}$$

sendo M_0 uma matriz simétrica constante, não singular, dada por $S(q)^T M(q) S(q)$ e $C_0(\dot{q}) = C_0(\dot{\alpha}) = C_0(\dot{q}_2) = S(q)^T C(q, \dot{q}) S(q) + S(q)^T M(q) \dot{S}(q)$. Note que nesta passagem desaparece a matriz de restrição que estava presente no termo $A^T \lambda$ da equação dinâmica, pois $S^T A^T = 0$ (A está no espaço nulo de S). Acrescentando um distúrbio de torque $w = [w_d \ w_e]^T$ e substituindo (2.4) na equação (3.2) segue que

$$\ddot{q}_2 = A(\dot{q}_2)\dot{q}_2 + B\tau + Bw, \quad (3.3)$$

sendo $A(\dot{q}_2) = -M_2^{-1}C_2(\dot{q}_2)$ e $B = M_2^{-1}$. Somando e subtraindo $\ddot{q}_2^d$ e $A(\dot{q}_2)\dot{q}_2^d$ em (3.3), (sendo que ‘d’ sobrescrito significa valor desejado) e definindo o estado como

$$\tilde{x} = \begin{bmatrix} \dot{\tilde{q}}_2 \\ \tilde{q}_2 \end{bmatrix} = \begin{bmatrix} \dot{\theta}_d - \dot{\theta}_d^d \\ \dot{\theta}_e - \dot{\theta}_e^d \\ \theta_d - \theta_d^d \\ \theta_e - \theta_e^d \end{bmatrix}$$

a representação quase-LPV, em espaço de estado, para o controle de acompanhamento de trajetória do RMR é dada por

$$\dot{\tilde{x}} = \begin{bmatrix} A(\dot{q}_2) & 0 \\ I & 0 \end{bmatrix} \tilde{x} + \begin{bmatrix} I \\ 0 \end{bmatrix} u + \begin{bmatrix} B \\ 0 \end{bmatrix} w \quad (3.4)$$

sendo

$$u = -\ddot{q}_2^d + A(\dot{q}_2)\dot{q}_2^d + B\tau,$$

ou

$$\tau = B^{-1}(\ddot{q}_2^d - A(\dot{q}_2)\dot{q}_2^d + u). \quad (3.5)$$

Considere agora a seguinte transformação de estado

$$T_0 \tilde{x} = \begin{bmatrix} T_{11} & T_{12} \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{\tilde{q}}_2 \\ \tilde{q}_2 \end{bmatrix}, \quad (3.6)$$

sendo $T_{11}, T_{12} \in \mathbb{R}^{2 \times 2}$ matrizes constantes a serem determinadas. Assume-se que a matriz T_{11} é diagonal, de maneira a facilitar escolhe-se $T_{11} = t_{11}I$. Aplicando a transformação de estado (3.6) em (3.4), obtém-se

$$\begin{aligned} \dot{\tilde{x}} &= T_0^{-1} T_0 \begin{bmatrix} A(\dot{q}_2) & 0 \\ I & 0 \end{bmatrix} T_0^{-1} T_0 \tilde{x} + T_0^{-1} T_0 \begin{bmatrix} I \\ 0 \end{bmatrix} u + T_0^{-1} T_0 \begin{bmatrix} B \\ 0 \end{bmatrix} w \\ &= T_0^{-1} \begin{bmatrix} A(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1} T_{12} \end{bmatrix} T_0 \tilde{x} \\ &\quad + T_0^{-1} \begin{bmatrix} T_{12} T_{11}^{-1} & A(\dot{q}_2) T_{12} - T_{12}^2 T_{11}^{-1} \\ 0 & 0 \end{bmatrix} T_0 \tilde{x} + T_0^{-1} T_0 \begin{bmatrix} I \\ 0 \end{bmatrix} u + T_0^{-1} T_0 \begin{bmatrix} B \\ 0 \end{bmatrix} w \\ &= T_0^{-1} \begin{bmatrix} A(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1} T_{12} \end{bmatrix} T_0 \tilde{x} + T_0^{-1} \begin{bmatrix} T_{12} \dot{\tilde{q}}_2 - A(\dot{q}_2) T_{12} \tilde{q}_2 - T_{11} \ddot{q}_2^d + T_{11} A(\dot{q}_2) \dot{q}^d + T_{11} B \tau \\ 0 \end{bmatrix} \\ &\quad + T_0^{-1} \begin{bmatrix} M_2^{-1} \\ 0 \end{bmatrix} T_{11} w \\ &= T_0^{-1} \begin{bmatrix} A(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1} T_{12} \end{bmatrix} T_0 \tilde{x} \\ &\quad + T_0^{-1} \begin{bmatrix} M_2^{-1} \\ 0 \end{bmatrix} T_{11} [-M_2(\ddot{q}_2^d - T_{11}^{-1} T_{12} \dot{\tilde{q}}_2) - C_2(\dot{q}_2)(\tilde{q}_2^d - T_{11}^{-1} T_{12} \tilde{q}_2) + \tau] \\ &\quad + T_0^{-1} \begin{bmatrix} M_2^{-1} \\ 0 \end{bmatrix} T_{11} w \end{aligned}$$

Definindo

$$A_T(\dot{q}_2) = T_0^{-1} \begin{bmatrix} -M_0^{-1}C_0(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1}T_{12} \end{bmatrix} T_0, \quad (3.7)$$

$$B_T = T_0^{-1} \begin{bmatrix} M_0^{-1} \\ 0 \end{bmatrix}, \quad (3.8)$$

$$d = M_0 T_{11} M_0^{-1} (w - \Delta F_0(x_e)), \quad (3.9)$$

$$x_e = \begin{bmatrix} q_2^T & \dot{q}_2^T & (q_2^d)^T & (\dot{q}_2^d)^T & (\ddot{q}_2^d)^T \end{bmatrix}, \quad (3.10)$$

$$F_0(x_e) = M_0(\ddot{q}_2^d - T_{11}^{-1}T_{12}\dot{\ddot{q}}_2) + C_0(\dot{q}_2)(\dot{q}_2^d - T_{11}^{-1}T_{12}\ddot{q}_2) \text{ e} \quad (3.11)$$

$$\Delta F_0(x_e) = \Delta M_0(\ddot{q}_2^d - T_{11}^{-1}T_{12}\dot{\ddot{q}}_2) + \Delta C_0(\dot{q}_2)(\dot{q}_2^d - T_{11}^{-1}T_{12}\ddot{q}_2). \quad (3.12)$$

obtem-se

$$\dot{\tilde{x}} = A_T(\dot{q}_2)\tilde{x} + B_T T_{11}(-F_0(x_e) + \tau) + B_T d. \quad (3.13)$$

3.2 Controle $\mathcal{H}_\infty$ não linear via representação quase-LPV

Com a equação em espaço de estado formulada na representação quase-LPV (o parâmetro ρ em função do estado, ou seja, $\rho = \rho(x)$ resultante do modelo dinâmico (2.9)), o controle $\mathcal{H}_\infty$ para sistemas LPV pode ser aplicado a RMRs, gerando um controlador não linear baseado na dinâmica. A lei de controle não linear apresentada a seguir é baseada em Desigualdades Matriciais Lineares (DMLs). Trata-se de uma lei de controle por realimentação de estado $u = F(\rho)x$ que estabiliza o sistema em malha fechada garantindo que um ganho $\mathcal{L}_2$ entre o distúrbio e a saída seja limitado por um nível de atenuação $\gamma > 0$.

3.2.1 Ganho $\mathcal{L}_2$ para sistemas não lineares variantes no tempo

Considere um sistema não linear variante no tempo com entrada de distúrbio afim $w \in \mathbb{R}^p$ e saída controlada $z \in \mathbb{R}^q$

$$\begin{aligned} \dot{x} &= f(x, t) + g(x, t)w, \\ z &= h(x, t) + k(x, t)w, \end{aligned} \quad (3.14)$$

sendo $f(0, t) = 0$ e $h(0, t) = 0$ para todo $t \in [0, T]$, e $x \in \mathbb{R}^n$ o estado. Assume-se que $f(x, t)$, $g(x, t)$, $h(x, t)$ e $k(x, t)$ são funções continuamente diferenciáveis em relação a x e contínuas em t . O sistema (3.14) possui ganho $\mathcal{L}_2 \leq \gamma$ no intervalo $[0, T]$ se

$$\int_0^T \|z(t)\|^2 dt \leq \gamma^2 \int_0^T \|w(t)\|^2 dt, \quad (3.15)$$

para todo $T \geq 0$ e todo $w \in \mathcal{L}_2(0, T)$ com o sistema iniciando em $x(0) = 0$. Para sistemas lineares invariantes no tempo, a condição de ganho $\mathcal{L}_2 \leq \gamma$ corresponde à condição da norma $\mathcal{H}_\infty$ da função de transferência entre a entrada de distúrbio e a saída controlada ser limitada por γ , ou seja, $\|T_{zw}(s)\|_\infty \leq \gamma$.

3.2.2 Síntese do controle $\mathcal{H}_\infty$ para sistemas LPV por realimentação do estado

O problema consiste no controle por realimentação de estado dependente de parâmetro para estabilizar o sistema em malha fechada e fazer a norma $\mathcal{L}_2$ menor que um nível de desempenho especificado γ . Considere o problema de síntese do controle por realimentação do estado

$$\begin{aligned} \dot{x} &= A(\rho(t))x + B_1(\rho(t))w + B_2(\rho(t))u \\ z_1 &= C_1(\rho(t))x \\ z_2 &= C_2(\rho(t))x + u \end{aligned} \quad (3.16)$$

sendo $x \in \mathbb{R}^n$ o estado, $u \in \mathbb{R}^{q_4}$ a entrada de controle, $w \in \mathbb{R}^p$ o distúrbio de entrada, $z_1 \in \mathbb{R}^{q_3}$ e $z_2 \in \mathbb{R}^{q_4}$ as saídas, $A(\cdot)$, $B_1(\cdot)$, $B_2(\cdot)$, $C_1(\cdot)$ e $C_2(\cdot)$ matrizes contínuas de dimensões apropriadas e $\rho(t) \in F_P^\nu$ definido por

$$F_P^\nu = \{\rho \in \mathcal{C}^1(\mathbb{R}^+, \mathbb{R}^m) : \rho(t) \in P, \|\dot{\rho}_i\| \leq \nu_i, i = 1, \dots, m\},$$

sendo $P \subset \mathbb{R}^m$ um conjunto compacto e $\nu = [\nu_1 \cdots \nu_m]^T$ com $\nu_i \geq 0$.

Definição 3.2.1 (*Wu, 1995*) *O problema de realimentação de estado dependente de parâmetro, para o sistema LPV (3.16), é resolvido se existir uma função $Z \in C^1(\mathbb{R}^s, S^{n \times n})$ e uma $F \in C^0(\mathbb{R}^s \times \mathbb{R}^s, \mathbb{R}^{n_u \times n})$ tais que para todo $\rho(t) \in P$ e $\|\dot{\rho}_i\| \leq \nu_i, i = 1, 2, \dots, s$, $Z(\rho) > 0$ e*

$$\begin{bmatrix} A_F^T(\rho, \dot{\rho})Z(\rho) + Z(\rho)A_F(\rho, \dot{\rho}) + \sum_{i=1}^s \left(\nu_i \frac{\partial Z}{\partial \rho_i} \right) + C^T(\rho, \dot{\rho})C(\rho, \dot{\rho}) & Z(\rho)B_1(\rho) \\ B_1^T(\rho)Z(\rho) & -\gamma^2 I \end{bmatrix} < 0, \quad (3.17)$$

sendo $A_F(\rho, \dot{\rho}) := A(\rho) + B_2(\rho)F(\rho, \dot{\rho})$, $C(\rho, \dot{\rho}) := C_1(\rho) + D_{12}F(\rho, \dot{\rho})$ e $D_{12} = [0 \ I]^T$.

Se o problema de realimentação de estado dependente de parâmetro (3.17) tem solução, então a lei de controle $u = F(\rho(t))x$ estabilizará exponencialmente o sistema em malha fechada e garantirá que a norma $\mathcal{L}_2$ induzida seja menor que γ . O seguinte teorema fornece uma condição de existência para o controlador de realimentação de estado expresso em DMLs para o sistema LPV em malha aberta (3.16).

Teorema 3.2.1 (Wu, 1995) *Dado um conjunto compacto $P \in \mathbb{R}^s$, um nível de desempenho $\gamma > 0$ e o sistema (3.16), o problema de realimentação de estado tem solução se e somente se existir uma função $X \in C^1(\mathbb{R}^s, S^{n \times n})$ tal que para todo $\rho \in P$, $X(\rho) > 0$ e*

$$\begin{bmatrix} E(\rho) & X(\rho)C_1^T(\rho) & B_1(\rho) \\ C_1(\rho)X(\rho) & -I & 0 \\ B_1^T(\rho) & 0 & -\gamma^2 I \end{bmatrix} < 0, \quad (3.18)$$

sendo

$$E(\rho) = - \sum_{i=1}^m \pm \left(\nu_i \frac{\partial X}{\partial \rho_i} \right) + \hat{A}(\rho)X(\rho) + X(\rho)\hat{A}(\rho)^T - B_2(\rho)B_2^T(\rho)$$

e $\hat{A}(\rho) = A(\rho) - B_2(\rho)C_2(\rho)$.

Este teorema fornece um ganho $F(\rho)$ de realimentação de estado tal que

$$u = -(B_2(\rho)^T X^{-1}(\rho) + C_2(\rho))x, \quad (3.19)$$

garante que o sistema em malha tenha ganho $\mathcal{L}_2 \leq \gamma$ para toda variação paramétrica $\rho(t) \in F_P^\nu$.

O resultado acima é uma generalização natural da teoria de controle $\mathcal{H}_\infty$ para sistemas lineares. Uma função de Lyapunov dependente de parâmetros na forma $V(x, t) = x^T(t)X^{-1}(\rho(t))x(t)$ é assumida. Como resultado, deve-se resolver as DMLs paramétricas (3.18) que é um problema de otimização convexo com dimensão infinita.

3.2.3 Considerações computacionais

Um esquema computacional prático (Wu, 1995; Wu et al., 1996; Huang e Jadbabaie, 1998; Siqueira e Terra, 2004) pode ser utilizado para resolver as desigualdades matriciais lineares presentes na análise e síntese dos problemas LPV. Por simplicidade, considere o problema de en-

contrar $X(\rho(t))$ na Equação (3.18). Primeiro, escolha um conjunto de funções $\mathcal{C}^1$, $\{f_i(\rho(t))\}_{i=1}^M$, como base para $X(\rho)$, ou seja,

$$X(\rho(t)) = \sum_{i=1}^M f_i(\rho(t))X_i, \quad (3.20)$$

sendo $X_i \in S^{n \times n}$ a matriz coeficiente para $f_i(\rho(t))$. Se $X(\rho(t))$ em (3.18) é substituída por (3.20), o problema de realimentação do estado transforma-se no seguinte problema de otimização

$$\min_{\{X_i\}_{i=1}^M} \gamma^2,$$

sujeito a

$$\begin{bmatrix} E^*(\rho) & \sum_{j=1}^M f_j(\rho)X_j C_1^T(\rho) & B_1(\rho) \\ C_1(\rho) \sum_{j=1}^M f_j(\rho)X_j & -I & 0 \\ B_1^T(\rho) & 0 & -\gamma^2 I \end{bmatrix} < 0, \quad (3.21)$$

$$\sum_{j=1}^M f_j(\rho)X_j > 0,$$

sendo

$$E^*(\rho) = - \sum_{i=1}^m \pm \left(\nu_i \sum_{j=1}^M \frac{\partial f_j}{\partial \rho_i} X_j \right) + \sum_{j=1}^M f_j(\rho) (\hat{A}(\rho)X_j + X_j \hat{A}(\rho)^T) - B_2(\rho)B_2^T(\rho).$$

Note que (3.21) são DMLs em termos das variáveis matriciais $\{X_i\}_{i=1}^M$ que devem ser satisfeitas para todo parâmetro $\rho(t)$ em P . Para resolver este problema de otimização de dimensão infinita, divide-se o conjunto de parâmetros P em N pontos $\{\rho_k\}_{k=1}^N$ em cada dimensão. Então calcula-se as DMLs acima para estes pontos. Desde que (3.18) consiste em 2^m vínculos, um total de $(2^m + 1)N^m$ desigualdades matriciais afins em termos das M variáveis matriciais $\{X_i\}$ devem ser resolvidas. E $\sum \pm(\cdot)$ indica que toda combinação $+(\cdot)$ e $-(\cdot)$ deve ser satisfeita. Uma aproximação da densidade de pontos particionados, N , que garante uma solução global das DMLs é dada em Wu (1995); Wu et al. (1996).

Este esquema computacional possui algumas limitações. O número de parâmetros considerados e o número de divisões N devem ser escolhidos tais que a solução seja alcançada em um número de iterações realizáveis. Outro problema é a falta de justificativa teórica na escolha das funções base para $X(\rho)$. Geralmente, escolhem-se funções similares às encontradas nas matrizes de estado $A(\rho(t))$ (Apkarian e Adams, 1998).

3.3 Controle $\mathcal{H}_\infty$ não linear via Teoria dos Jogos (TJ)

Nesta seção, o controle $\mathcal{H}_\infty$ não linear proposto em Chen et al. (1994) é aplicado aos RMRs onde se leva em consideração a equação em espaço de estado da dinâmica do erro de ordem reduzida (3.4). O termo de incertezas $\Delta F_0(x_e)$, associado a um distúrbio no torque w , forma o distúrbio combinado d definido em (3.13). Escolhendo a entrada de controle como sendo $u = T_{11}(-F_0(x_e) + \tau)$, o torque aplicado pode ser calculado como

$$\tau = F_0(x_e) + T_{11}^{-1}u. \quad (3.22)$$

Considerando a dinâmica do erro de acompanhamento, (3.13), o critério de desempenho que inclui um nível de atenuação de distúrbio desejado γ é definido como

$$\min_{u(\cdot) \in \mathcal{L}_2} \max_{0 \neq d(\cdot) \in \mathcal{L}_2} \frac{\int_0^\infty \left(\frac{1}{2} \tilde{x}^T(t) Q \tilde{x}(t) + \frac{1}{2} u^T(t) R u(t) \right) dt}{\int_0^\infty \left(\frac{1}{2} d^T(t) d(t) \right) dt} \leq \gamma^2, \quad (3.23)$$

sendo Q e R matrizes de ponderação simétricas definidas positivas e $\tilde{x}(0) = 0$. Este critério de desempenho é semelhante ao apresentado em (3.15), sendo que neste caso matrizes de ponderação são acrescentadas no estado e na entrada de controle.

A solução do problema de controle $\mathcal{H}_\infty$ (3.23) relacionado à equação de estado (3.4), pode ser explicitamente encontrada pela teoria dos jogos diferenciais (Basar e Bernhard, 1990; Basar e Olsder, 1982) e uma escolha apropriada da função de Lyapunov $V(\tilde{x}, t)$ (Chen et al., 1994).

A metodologia apresentada por Chen et al. (1994) para resolver este problema é resumida a seguir. O critério de desempenho (3.23) pode ser modificado para formar o seguinte problema *minimax*

$$\min_{u(\cdot) \in \mathcal{L}_2} \max_{0 \neq d(\cdot) \in \mathcal{L}_2} \int_0^\infty \left(\frac{1}{2} \tilde{x}^T(t) Q \tilde{x}(t) + \frac{1}{2} u^T(t) R u(t) - \frac{1}{2} \gamma^2 d^T(t) d(t) \right) dt \leq 0,$$

com $\tilde{x}(0) = 0$. Definindo a função custo

$$J(\tilde{x}(t), u, d, t) = \int_t^\infty L(\tilde{x}(s), u(s), d(s)) ds,$$

sendo $L(\tilde{x}, u, d)$ o Lagrangiano dado por

$$L(\tilde{x}, u, d) = \frac{1}{2} \tilde{x}^T(t) Q \tilde{x}(t) + \frac{1}{2} u^T(t) R u(t) - \frac{1}{2} \gamma^2 d^T(t) d(t).$$

Definindo-se a função de Lyapunov

$$V(\tilde{x}(t), t) = \min_{u(\cdot)} \max_{d(\cdot)} J(\tilde{x}(t), u, d, t),$$

o critério de desempenho (3.23) fica

$$V(\tilde{x}(0), 0) = \min_{u(\cdot)} \max_{d(\cdot)} J(\tilde{x}(0), u, d, 0) \leq 0,$$

com $\tilde{x}(0) = 0$. De acordo com a teoria dos jogos diferenciais, a solução deste problema *minimax* é encontrada se existir uma função de Lyapunov continuamente diferenciável $V(\tilde{x}, t)$ que satisfaz a seguinte equação *minimax* de Bellman-Isaacs

$$-\frac{\partial V(\tilde{x}, t)}{\partial t} = \min_{u(\cdot)} \max_{d(\cdot)} \left\{ L(\tilde{x}, u, d) + \left(\frac{\partial V(\tilde{x}, t)}{\partial \tilde{x}} \right)^T \dot{\tilde{x}} \right\},$$

com condição terminal $V(\tilde{x}(\infty), \infty) = 0$. Escolhendo a função de Lyapunov da forma

$$V(\tilde{x}, t) = \frac{1}{2} \tilde{x}^T P(\tilde{x}, t) \tilde{x}, \quad (3.24)$$

sendo P uma matriz simétrica definida positiva para todo t , a equação de Bellman-Isaacs fornece a seguinte equação de Riccati

$$\dot{P} + PA_T + A_T^T P - PB_T \left(R^{-1} - \frac{1}{\gamma^2} T_{11}^2 \right) B_T^T P + Q = 0.$$

O controle ótimo correspondente e o pior caso de distúrbio são dados, respectivamente, por

$$u^* = -R^{-1} B_T^T P \tilde{x}$$

e

$$d^* = \frac{1}{\gamma^2} B_T^T P \tilde{x}.$$

Com uma escolha apropriada da matriz P e sendo a matriz $(C_0(\dot{q}_2) - \frac{1}{2}\dot{M}_0)$ anti-simétrica (Chen et al., 1994), a equação de Riccati pode ser simplificada para uma equação matricial algébrica. De acordo com Chen et al. (1994); Johansson (1990) uma escolha apropriada para P é feita em função de M_0 , T_0 e uma matriz K simétrica definida positiva a ser determinada, ou

seja,

$$P = T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0. \quad (3.25)$$

A equação algébrica simplificada é dada por

$$\begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} - T_0^T [I \ 0]^T \left(R^{-1} - \frac{1}{\gamma^2} T_{11}^2 \right) [I \ 0] T_0 + Q = 0. \quad (3.26)$$

O controle ótimo e o pior distúrbio podem ser reescritos, respectivamente como

$$u^* = -R^{-1} [I \ 0] T_0 \tilde{x} \quad (3.27)$$

e

$$d^* = \frac{T_{11}}{\gamma^2} [I \ 0] T_0 \tilde{x}.$$

A condição terminal é satisfeita para esta escolha de P , (Chen et al., 1994).

3.4 Controle $\mathcal{H}_\infty$ não linear baseado em modelo fuzzy Takagi Sugeno

Nesta seção, duas abordagens de controle robusto baseadas em lógica fuzzy serão aplicadas em RMRs, seguindo as linhas desenvolvidas em Ge et al. (1998); Chang (2000, 2005). Duas hipóteses serão consideradas: na primeira, os termos das equações (3.11) e (3.12) $F_0(x_e)$ e $\Delta F_0(x_e)$ são considerados desconhecidos e na segunda $F_0(x_e)$ é considerado conhecido e apenas as incertezas $\Delta F_0(x_e)$ são consideradas desconhecidas.

Para o primeiro caso, onde os termos $F_0(x_e)$ e $\Delta F_0(x_e)$ do modelo do RMR são considerados desconhecidos, um modelo fuzzy T-S $\hat{F}(x_e, \Theta)$, formado por um conjunto de modelos fuzzy, é proposto para estimar ambos os termos.

Em geral, um sistema fuzzy consiste de quatro partes: um fuzzificador, uma base de regras, um procedimento de inferência e um defuzzificador. O fuzzificador é um mapeamento do universo de discurso de entrada $U \subset \mathbb{R}^r$ dos conjuntos fuzzy definidos em U . Dois fatores determinam a interface de fuzzificação: (i) o número de conjuntos fuzzy definidos no universo de discurso de entrada e (ii) as funções de pertinência relacionadas com esses conjuntos.

A base de regras é formada por um conjunto de proposições linguísticas do tipo

SE premissas são satisfeitas,

ENTÃO conseqüências são inferidas.

O método de T-S é caracterizado por uma base de regras que utiliza conseqüentes funcionais ao invés de conseqüentes fuzzy, na forma

SE u_1 é A_{11} e u_2 é A_{12} e ... e u_r é A_{1r} ,

ENTÃO $y_1 = p_{10} + p_{11}u_1 + p_{12}u_2 + \dots + p_{1r}u_r$.

$\vdots$

SE u_1 é A_{k1} e u_2 é A_{k2} e ... e u_r é A_{kr} ,

ENTÃO $y_k = p_{k0} + p_{k1}u_1 + p_{k2}u_2 + \dots + p_{kr}u_r$.

sendo A_{ij} , $j = 1, \dots, r$ e $i = 1, \dots, k$, variáveis linguísticas referentes aos conjuntos fuzzy definidos sobre os espaços de entrada $U_1, U_2, \dots, U_r$; $u_1, u_2, \dots, u_r$ valores das variáveis de entrada e k o número de regras fuzzy.

O procedimento de inferência é a lógica da tomada de decisão que aplica as regras fuzzy para determinar a saída correspondente às entradas fuzzificadas. A saída inferida pelo sistema fuzzy através do método de T-S é *crisp* (portanto não necessita de um defuzzificador), definida pela média ponderada das saídas y_i de cada subsistema linear

$$y = \frac{\sum_{i=1}^k \mu_i y_i}{\sum_{i=1}^k \mu_i} = \psi \theta, \quad (3.28)$$

sendo

$$\psi = [\beta_1, \dots, \beta_k, u_1\beta_1, \dots, u_1\beta_k, \dots, u_r\beta_k],$$

$$\beta_i = \frac{\mu_i}{\sum_{i=1}^k \mu_i},$$

$$\theta = [p_{10}, \dots, p_{k0}, p_{11}, \dots, p_{k1}, \dots, p_{kr}]$$

os parâmetros os quais definem as funções de conseqüências $y_i = f(\tilde{x})$ para cada regra e μ_i é o grau de liberdade da i -ésima regra, definido como o mínimo entre os graus de pertinência associados às entradas nos conjuntos fuzzy ativados pela i -ésima regra:

$$\mu_i := A_{i1}(u_1) \wedge A_{i2}(u_2) \wedge \dots \wedge A_{ir}(u_r). \quad (3.29)$$

O modelo fuzzy T-S, $\hat{F}(x_e, \Theta)$, proposto para estimar $F_0(x_e)$ e $\Delta F_0(x_e)$, é definido como

segue

$$\begin{aligned}
\hat{F}(x_e, \Theta) &= Y(\tilde{x}, A(\tilde{x}), \Theta) = [y^1, \dots, y^n]^T \\
&= \begin{bmatrix} \hat{F}_1(x_e, \theta_1) \\ \vdots \\ \hat{F}_n(x_e, \theta_n) \end{bmatrix} = \begin{bmatrix} \psi_1 \theta_1 \\ \vdots \\ \psi_n \theta_n \end{bmatrix} \\
&= \begin{bmatrix} \psi_1 & 0 & \dots & 0 \\ 0 & \psi_2 & \vdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \psi_n \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} \\
&= \Psi(x_e)(\tilde{x}, A(\tilde{x}))\Theta,
\end{aligned} \tag{3.30}$$

sendo Y o conjunto de sistemas fuzzy, $\tilde{x}$ são as entradas fuzzy, $A(\tilde{x}) := [A_1(\tilde{q}) \ A_2(\tilde{q})]$ representa os conjuntos fuzzy definidos para as entradas de fuzzificação e n o número de atuadores robóticos, que para o caso do RMR são dois (atuadores da roda direita e esquerda).

Considere uma região de restrição Ω_Θ de parâmetros Θ . Para garantir que o parâmetro estimado $\hat{\Theta}(t)$ para todo t esteja dentro da região de restrição, um algoritmo de projeção pode ser usado. Considere $\Omega_0 = \{\Theta : \Theta^T \Theta \leq N\}$ e $\Omega_\Theta = \Theta : \{\Theta^T \Theta \leq N + \Delta\}$ para algum $N > 0$ e $\Delta > 0$. De acordo com Chang (2000), define-se

$$Proj[\Phi] = \begin{cases} \Phi, & \text{se } \Theta^T \Theta \leq N \text{ ou} \\ & (\Theta^T \Theta > N \text{ e } \Theta^T \Phi \leq 0) \\ \Phi - \frac{(\Theta^T \Theta - N)\Theta^T \Phi}{\delta \Theta^T \Theta} \Theta, & \text{caso contrário} \end{cases} \tag{3.31}$$

para alguma função suave $\Phi(x_e)$. Assume-se que exista um valor $\Theta^* \in \Omega_0$ tal que $\hat{F}(x_e, \Theta^*)$ pode aproximar $F(x_e)$ o mais próximo possível. Um modo de alcançar o desempenho $\mathcal{H}_\infty$, quando a rede neural é usada, é considerar o erro de aproximação, $\Delta F(x_e) = \hat{F}(x_e, \Theta^*) - F(x_e)$, como distúrbio. Essa abordagem foi usada em Chang e Chen (1997), onde o termo dinâmico do robô $F(x_e)$ é aproximado por uma rede neural.

Em virtude do erro de aproximação incluir os efeitos de incertezas dinâmicas, a integração do termo quadrático na prova da estabilidade para a garantia do critério $\mathcal{H}_\infty$ se torna mais complexa. Em Chang (2000) a solução desse problema tem sido obtida com a utilização da técnica

de controle por estrutura variável (CEV), cujos detalhes podem ser vistos nessa referência.

A utilização do CEV requer que o erro de aproximação deva ser limitado por uma função dependente do estado, isto é, existe uma função $k(x_e) > 0$ tal que $|(\Delta F(x_e))_i| \leq k(x_e)$, para todo $1 \leq i \leq n$.

Levando em conta o erro de aproximação mínimo $\Delta F(x_e)$, a equação dinâmica do erro com respeito a $\tilde{x}$ pode ser obtida como

$$\dot{\tilde{x}} = A_T(\dot{q}_2)\tilde{x} + B_T T_{11}(-\Psi(x_e)\Theta^* + \Delta F(x_e) + \tau) + B_T d, \quad (3.32)$$

com $d = M_0 T_{11} M_0^{-1} w$. O próximo teorema apresenta uma síntese de um controlador adaptativo $\mathcal{H}_\infty$ baseado em modelo fuzzy T-S, considerando a estimativa completa do modelo do RMR. A prova do teorema abaixo é similar ao apresentado em Chang (2000).

Teorema 3.4.1 *Considere o sistema robótico descrito em (2.9) com $d \in \mathcal{L}_2[0, \infty)$. Dado um nível de atenuação $\gamma > 0$ e uma matriz de ponderação $Q = Q^T > 0$, se existirem uma matriz simétrica definida positiva $K = K^T > 0$ e uma matriz T_0 satisfazendo a equação algébrica de Riccati*

$$\begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} - T_0^T [I \ 0]^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) [I \ 0] T_0 + Q = 0 \quad (3.33)$$

sendo R a matriz de ganho tal que $R < \gamma^2 I$, então a lei de controle adaptativo $\mathcal{H}_\infty$ baseado em modelo fuzzy T-S é dada por

$$\tau = \Psi(x_e)\Theta + T_{11}^{-1}u + u_s \quad (3.34)$$

com

$$\dot{\Theta} = Proj[-H^{-1}\Psi(x_e)^T T_{11}[I \ 0]T_0\tilde{x}], \quad (3.35)$$

$$u = -R^{-1}[I \ 0]T_0\tilde{x}, \quad (3.36)$$

$$u_s = -k(x_e)sgn(T_{11}[I \ 0]T_0\tilde{x}), \quad (3.37)$$

sendo H o ganho adaptativo e $sgn(T_{11}[I \ 0]T_0\tilde{x}) \triangleq [sgn((T_{11}[I \ 0]T_0\tilde{x})_1) \dots sgn((T_{11}[I \ 0]T_0\tilde{x})_n)]^T$. Esta lei de controle garante que todas as variáveis do sistema em malha fechada (2.9, 3.34-3.37) são limitadas e que a seguinte desigualdade é satisfeita

$$\int_0^T (\tilde{x}^T Q \tilde{x} + u^T R u) dt \leq \tilde{x}^T(0) P_0 \tilde{x}(0) + \tilde{\Theta}^T(0) H \tilde{\Theta}(0) + \gamma^2 \int_0^T (d^T d) dt.$$

Prova 1 Considere a seguinte equação de Lyapunov

$$V(\tilde{x}, \tilde{\Theta}, t) = \frac{1}{2} \tilde{x}^T P \tilde{x} + \frac{1}{2} \tilde{\Theta}^T H \tilde{\Theta} \quad (3.38)$$

Tomando a derivada de V em relação ao tempo, obtém-se

$$\dot{V}(\tilde{x}, \tilde{\Theta}, t) = \frac{1}{2} \tilde{x}^T \dot{P} \tilde{x} + \frac{1}{2} \dot{\tilde{x}}^T P \tilde{x} + \frac{1}{2} \tilde{x}^T P \dot{\tilde{x}} + \frac{1}{2} \dot{\tilde{\Theta}}^T S \tilde{\Theta} + \frac{1}{2} \tilde{\Theta}^T S \dot{\tilde{\Theta}}. \quad (3.39)$$

Substituindo (3.32) em (3.39), tem-se

$$\begin{aligned} \dot{V}(\tilde{x}, \tilde{\Theta}, t) &= \frac{1}{2} \tilde{x}^T \dot{P} \tilde{x} + \frac{1}{2} \tilde{x}^T A_T^T(\dot{q}_2) + [B_T T_{11}(-\Psi(x_e)\Theta^* + \Delta F(x_e) + \tau) + B_T d]^T P \tilde{x} \\ &+ \frac{1}{2} \tilde{x}^T P [A_T(\dot{q}_2) \tilde{x} + B_T T_{11}(-\Psi(x_e)\Theta^* + \Delta F(x_e) + \tau) + B_T d] + \dot{\tilde{\Theta}}^T S \tilde{\Theta}. \end{aligned} \quad (3.40)$$

Usando o controlador (3.34) e definindo $\tilde{\Theta} \triangleq \Theta - \Theta^*$, a derivada de $\dot{V}$ torna-se

$$\begin{aligned} \dot{V}(\tilde{x}, \tilde{\Theta}, t) &= \frac{1}{2} \tilde{x}^T (\dot{P} + A_T^T(\dot{q}_2) P + P A_T(\dot{q}_2)) \tilde{x} + \tilde{x}^T P B_T (u + d) \\ &+ \tilde{x}^T P B_T T_{11} (u_s + \Delta F(x_e)) + \tilde{x}^T P B_T T_{11} \Psi(x_e) \tilde{\Theta} + \dot{\tilde{\Theta}}^T H \tilde{\Theta}. \end{aligned} \quad (3.41)$$

Considerando $N = (C_0(\dot{q}_2) - \frac{1}{2} \dot{M}_0)$ anti-simétrica para todo t , a matriz P definida como em (3.25) e manipulando o termo $A_T^T(\dot{q}_2) P + P A_T(\dot{q}_2)$ de (3.41), obtém-se

$$\begin{aligned} A_T^T(\dot{q}_2) P + P A_T(\dot{q}_2) &= T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0 T_0^{-1} \begin{bmatrix} -M_0^{-1} C_0(\dot{q}_2) & 0 \\ T_{11}^{-1} & -T_{11}^{-1} T_{12} \end{bmatrix} T_0 \\ &+ T_0^T \begin{bmatrix} -M_0^{-1} C_0(\dot{q}_2) & T_{11}^{-1} \\ 0 & -T_{11}^{-1} T_{12} \end{bmatrix} T_0^{-T} T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0 \\ &= T_0^T \begin{bmatrix} -C_0(\dot{q}_2) & 0 \\ K T_{11}^{-1} & -K T_{11}^{-1} T_{12} \end{bmatrix} T_0 + T_0^T \begin{bmatrix} -C_0(\dot{q}_2) & K T_{11}^{-1} \\ 0 & -K T_{11}^{-1} T_{12} \end{bmatrix} T_0 \\ &= T_0^T \begin{bmatrix} -2C_0(\dot{q}_2) & K T_{11}^{-1} \\ K T_{11}^{-1} & -2K T_{11}^{-1} T_{12} \end{bmatrix} T_0 \\ &= T_0^T \begin{bmatrix} 0 & K T_{11}^{-1} \\ K T_{11}^{-1} & -2K T_{11}^{-1} T_{12} \end{bmatrix} T_0 + T_0^T \begin{bmatrix} -2C_0(\dot{q}_2) & 0 \\ 0 & 0 \end{bmatrix} T_0 \\ &= \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} + T_0^T \begin{bmatrix} -\dot{M}_0 & 0 \\ 0 & 0 \end{bmatrix} T_0 + T_0^T \begin{bmatrix} -2N & 0 \\ 0 & 0 \end{bmatrix} T_0. \end{aligned} \quad (3.42)$$

Manipulando o termo PB_T , tem-se

$$\begin{aligned} PB_T &= T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0 T_0^{-1} \begin{bmatrix} M_0^{-1} \\ 0 \end{bmatrix} \\ &= T_0^T [I \ 0]^T. \end{aligned} \quad (3.43)$$

Substituindo (3.42) e (3.43) em (3.41), $\dot{V}$ torna-se

$$\begin{aligned} \dot{V} &= \frac{1}{2} \tilde{x}^T \left\{ T_0^T \begin{bmatrix} \dot{M}_0 & 0 \\ 0 & 0 \end{bmatrix} T_0 + \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} + T_0^T \begin{bmatrix} -\dot{M}_0 & 0 \\ 0 & 0 \end{bmatrix} T_0 + T_0^T \begin{bmatrix} -2N & 0 \\ 0 & 0 \end{bmatrix} T_0 \right\} \tilde{x} \\ &+ \tilde{x}^T T_0^T [I \ 0]^T (u + d) + \tilde{x}^T T_0^T [I \ 0]^T T_{11} (u_s + \Delta F(x_e)) \\ &+ \tilde{x}^T T_0^T [I \ 0]^T T_{11} \Psi(x_e) \tilde{\Theta} + \dot{\tilde{\Theta}}^T H \tilde{\Theta}. \end{aligned} \quad (3.44)$$

Como a matriz N é anti-simétrica, o termo quadrático que inclui a matriz N é zero, desse modo (3.44) torna-se

$$\begin{aligned} \dot{V}(\tilde{x}, \tilde{\Theta}, t) &= \frac{1}{2} \tilde{x}^T \begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} \tilde{x} + \tilde{x}^T T_0^T [I \ 0]^T (u + d) + \tilde{x}^T T_0^T [I \ 0]^T T_{11} (u_s + \Delta F(x_e)) \\ &+ \tilde{x}^T T_0^T [I \ 0]^T T_{11} \Psi(x_e) \tilde{\Theta} + \dot{\tilde{\Theta}}^T H \tilde{\Theta}. \end{aligned} \quad (3.45)$$

Avaliando o termo $\tilde{x}^T T_0^T [I \ 0]^T T_{11} (u_s + \Delta F(x_e))$ e a lei de controle (3.37), observa-se que

$$\begin{aligned} \tilde{x}^T T_0^T [I \ 0]^T T_{11} u_s &= -\tilde{x}^T T_0^T [I \ 0]^T T_{11} k(x_e) \text{sgn}(T_{11} [I \ 0] T_0 \tilde{x}) \\ &= -[(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 + \dots + (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n] k(x_e) \text{sgn} \begin{bmatrix} (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 \\ \vdots \\ (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n \end{bmatrix} \\ &= -k(x_e) [(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 \text{sgn}((\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1) + \dots \\ &+ (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n \text{sgn}((\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n)] \\ &= -k(x_e) [|(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| + \dots + |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n|] \\ &= -k(x_e) \sum_{p=1}^n |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_p|, \end{aligned} \quad (3.46)$$

e que

$$\begin{aligned}
\tilde{x}^T T_0^T [I \ 0]^T T_{11} \Delta F(x_e) &= [(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 \dots (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n] \begin{bmatrix} (\Delta F(x_e))_1 \\ \vdots \\ (\Delta F(x_e))_n \end{bmatrix} \\
&= (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 (\Delta F(x_e))_1 + \dots + (\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n (\Delta F(x_e))_n \\
&\leq |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1 (\Delta F(x_e))_1| + \dots + |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n (\Delta F(x_e))_n| \\
&\leq |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| |(\Delta F(x_e))_1| + \dots + |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n| |(\Delta F(x_e))_n| \\
&= |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| |(\Delta F(x_e))_1| + \dots + |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n| |(\Delta F(x_e))_n| \\
&= \sum_{p=1}^n |(\Delta F(x_e))_p| |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_p| \tag{3.47}
\end{aligned}$$

Levando-se em consideração (3.46) e (3.47), obtém-se a seguinte desigualdade

$$\begin{aligned}
&\tilde{x}^T T_0^T [I \ 0]^T T_{11} (u_s + \Delta F(x_e)) \tag{3.48} \\
&\leq -k(x_e) \sum_{i=1}^n |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_i| + \sum_{i=1}^n |(\Delta F(x_e))_i| |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_i| \\
&= -k(x_e) |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| + |(\Delta F(x_e))_1| |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| + \dots + \\
&\quad - k(x_e) |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n| + |(\Delta F(x_e))_n| |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n| \\
&= [|(\Delta F(x_e))_1| - k(x_e)] |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_1| + \dots \\
&\quad + [|(\Delta F(x_e))_n| - k(x_e)] |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_n| \tag{3.49}
\end{aligned}$$

Levando em consideração a suposição de CEV, $|(\Delta F(x_e))_p| \leq k(x_e)$ para todo $1 \leq p \leq n$, então pode se garantir que

$$\begin{aligned}
&\tilde{x}^T T_0^T [I \ 0]^T T_{11} (u_s + \Delta F(x_e)) \\
&\leq -k(x_e) \sum_{p=1}^n |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_p| + \sum_{p=1}^n |(\Delta F(x_e))_p| |(\tilde{x}^T T_0^T [I \ 0]^T T_{11})_p| \leq 0. \tag{3.50}
\end{aligned}$$

Substituindo (3.33) e (3.50) em (3.45), obtém-se

$$\begin{aligned}\dot{V}(\tilde{x}, \tilde{\Theta}, t) = & -\frac{1}{2}\tilde{x}^T Q \tilde{x} - \frac{1}{2}u^T R u - \frac{1}{2}\tilde{x}^T T_0^T [I \ 0]^T \frac{1}{\gamma^2} [I \ 0] T_0 \tilde{x} + \tilde{x} [I \ 0]^T d \\ & + \tilde{x}^T T_0^T [I \ 0]^T \Psi(x_e) \tilde{\Theta} + \dot{\tilde{\Theta}}^T H \tilde{\Theta}.\end{aligned}\quad (3.51)$$

Somando e subtraindo $\frac{1}{2}\gamma^2 d^T d$ em 3.51 e sabendo que

$$\begin{aligned}-\frac{1}{2}\left(\frac{1}{\gamma}[I \ 0]T_0 \tilde{x} - \gamma d\right)^T \left(\frac{1}{\gamma}[I \ 0]T_0 \tilde{x} - \gamma d\right) = & -\frac{1}{2}\tilde{x}^T T_0^T [I \ 0]^T \frac{1}{\gamma^2} [I \ 0] T_0 \tilde{x} \\ & + \tilde{x}^T T_0^T [I \ 0]^T d - \frac{1}{2}\gamma^2 d^T d,\end{aligned}\quad (3.52)$$

tém-se

$$\begin{aligned}\dot{V}(\tilde{x}, \tilde{\Theta}, t) \leq & -\frac{1}{2}\left(\frac{1}{\gamma}[I \ 0]T_0 \tilde{x} - \gamma d\right)^T \left(\frac{1}{\gamma}[I \ 0]T_0 \tilde{x} - \gamma d\right) \\ & - \frac{1}{2}\tilde{x}^T Q \tilde{x} - \frac{1}{2}u^T R u + \frac{1}{2}\gamma^2 d^T d + \tilde{x}^T T_0^T [I \ 0]^T T_{11} \Psi(x_e) \tilde{\Theta} + \dot{\tilde{\Theta}}^T H \tilde{\Theta}.\end{aligned}\quad (3.53)$$

Desde que $\dot{\tilde{\Theta}} = \dot{\Theta}$ e usando o algoritmo de projeção

$$\dot{V}(\tilde{x}, \tilde{\Theta}, t) \leq -\frac{1}{2}\tilde{x}^T Q \tilde{x} - \frac{1}{2}u^T R u + \frac{1}{2}\gamma^2 d^T d. \quad (3.54)$$

Integrando a inequação acima de $t = 0$ para $t = T$, resulta

$$\begin{aligned}V(\tilde{x}(T), \tilde{\Theta}(T), T) - V(\tilde{x}(0), \tilde{\Theta}(0), 0) \\ \leq -\frac{1}{2}\int_0^T (\tilde{x}^T Q \tilde{x}) dt - \frac{1}{2}\int_0^T (u^T R u) dt + \frac{\gamma^2}{2}\int_0^T (d^T d) dt.\end{aligned}\quad (3.55)$$

Desde que $V(x(\tilde{T}), \Theta(\tilde{T}), T) \geq 0$, a inequação acima resulta

$$\int_0^T (\tilde{x}^T Q \tilde{x} + u^T R u) dt \leq \tilde{x}^T(0) P_0 \tilde{x}(0) + \tilde{\Theta}^T(0) H \tilde{\Theta}(0) + \gamma^2 \int_0^T (d^T d) dt. \quad (3.56)$$

Além disso, desde que $d \in \mathcal{L}_2[0, \infty)$, ou seja, existe uma constante finita $M_d > 0$ tal que

$\int_0^T (d^T d) dt \leq M_d$, a inequação (3.55) também implica que

$$V(\tilde{x}(T), \tilde{\Theta}(T), T) \leq V(\tilde{x}(0), \tilde{\Theta}(0), 0) + \frac{1}{2} \gamma^2 M_d < \infty. \quad (3.57)$$

Isto implica que todos os estados e sinais do sistema de malha fechada são limitados.

◇

Se considerarmos agora a hipótese de que se conhece o modelo nominal do RMR e apenas as incertezas do robô serão estimadas pelo modelo fuzzy T-S, a equação dinâmica do erro passa a ser escrita da seguinte forma

$$\dot{\tilde{x}} = A_T(\dot{q}_2) \tilde{x} + B_T T_{11} (-F_0(x_e) - \Psi(x_e) \Theta^* + \Delta F(x_e) + \tau) + B_T d, \quad (3.58)$$

com $d = M_0(q) T_{11} M_0^{-1}(q) w$. A lei de controle adaptativo $\mathcal{H}_\infty$ baseado no modelo matemático do sistema nominal do RMR e no modelo fuzzy T-S é semelhante ao controlador apresentado no Teorema 3.4.1 sendo que apenas o torque do controlador é reescrito da seguinte forma

$$\tau = F_0(x_e) + \Psi(x_e) \Theta + T_{11}^{-1} u + u_s. \quad (3.59)$$

Perceba que o modelo fuzzy T-S apenas complementa o modelo nominal $F_0(x_e)$.

3.5 Controle $\mathcal{H}_\infty$ não linear baseado em redes neurais

Nesta seção, as duas abordagens de controle robusto utilizadas na seção anterior baseadas em lógica fuzzy serão agora aplicadas em RMRs com base em redes neurais, seguindo as linhas desenvolvidas em Ge et al. (1998); Chang (2000, 2005). Ambas as hipóteses serão adotadas novamente: primeiro consideram-se os termos das equações (3.11) e (3.12) $F_0(x_e)$ e $\Delta F_0(x_e)$ desconhecidos e em seguida $F_0(x_e)$ é considerado conhecido e apenas as incertezas $\Delta F_0(x_e)$ são consideradas desconhecidas.

Para o primeiro problema, onde os termos $F_0(x_e)$ e $\Delta F_0(x_e)$ do modelo do RMR são considerados desconhecidos, uma rede neural $\hat{F}(x_e, \Theta)$, formada por um conjunto de redes neurais, é proposta para estimar ambos os termos.

Definem-se n redes neurais $\xi_k^T \Theta_k$, $k = 1, \dots, n$, compostas de neurônios não lineares em todas as camadas escondidas e neurônios lineares nas camadas de entrada e saída, com parâmetros

ajustáveis Θ_k na camada de saída (Chang, 2000; Chang e Chen, 1997), Fig. 3.1. As saídas das redes neurais são da forma

$$\xi_k^T \Theta_k = \sum_{i=1}^{p_k} H \left(\sum_{j=1}^{5n} w_{ij}^k x_{ej} + m_i^k \right) \Theta_{ki} \quad (3.60)$$

com

$$\xi_k = \begin{bmatrix} H \left(\sum_{j=1}^{5n} w_{1j}^k x_{ej} + b_1^k \right) \\ \vdots \\ H \left(\sum_{j=1}^{5n} w_{p_k j}^k x_{ej} + b_{p_k}^k \right) \end{bmatrix}, \Theta_k = \begin{bmatrix} \Theta_{k1} \\ \vdots \\ \Theta_{kp_k} \end{bmatrix}$$

sendo p_k o número de neurônios da camada escondida. Os pesos w_{ij}^k e o bias b_i^k para $1 \leq i \leq p_k$, $1 \leq j \leq 5n$ e $1 \leq k \leq n$ são assumidos constantes e especificados pelo projetista e $H(\cdot)$ é escolhida como sendo uma função tangente hiperbólica

$$H(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}.$$

Desse modo, a rede neural completa pode ser denotada por

$$\begin{aligned} \hat{F}(x_e, \Theta) &= \begin{bmatrix} \hat{F}_1(x_e, \Theta_1) \\ \vdots \\ \hat{F}_n(x_e, \Theta_n) \end{bmatrix} = \begin{bmatrix} \xi_1 \Theta_1 \\ \vdots \\ \xi_n \Theta_n \end{bmatrix} \\ &= \begin{bmatrix} \xi_1 & 0 & \dots & 0 \\ 0 & \xi_2 & \vdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \xi_n \end{bmatrix} \begin{bmatrix} \Theta_1 \\ \Theta_2 \\ \vdots \\ \Theta_n \end{bmatrix} = \Xi \Theta. \end{aligned} \quad (3.61)$$

Considere o algoritmo de projeção (3.31) e assuma que exista um valor de parâmetro $\Theta^* \in \Omega_\Theta$ tal que $\hat{F}(x_e, \Theta^*)$ pode aproximar $F(x_e)$ o mais próximo possível. Seja $\Delta F(x_e) = \hat{F}(x_e, \Theta^*) - F(x_e)$, sem perda de generalidade assuma que exista uma função $k(x_e) > 0$ tal que $|(\Delta F(x_e))_p| \leq k(x_e)$, para todo $1 \leq p \leq n$.

Com essas suposições, o problema de controle $\mathcal{H}_\infty$ não linear baseado em redes neurais para sistemas robóticos pode ser formulado como segue: dado um nível de atenuação γ , encontre um

Figura 3.1: Rede Neural.

controlador adaptativo baseado em redes neurais com realimentação de estado

$$\tau = \Xi\Theta + T_{11}^{-1}u + u_s \quad (3.62)$$

com

$$\dot{\Theta} = Proj[-Z^{-1}\Xi^T T_{11}[I \ 0]T_0\tilde{x}] \quad (3.63)$$

$$u = -R^{-1}[I \ 0]T_0\tilde{x} \quad (3.64)$$

$$u_s = -k(x_e)sgn(T_{11}[I \ 0]T_0\tilde{x}) \quad (3.65)$$

tal que o seguinte índice de desempenho seja satisfeito

$$\int_0^T (\tilde{x}^T Q \tilde{x} + u^T R u) dt \leq \tilde{x}^T(0)P\tilde{x}(0) + \tilde{\Theta}^T(0)Z\tilde{\Theta}(0) + \gamma^2 \int_0^T (d^T d)dt, \quad (3.66)$$

sendo $Q = Q^T > 0, R = R^T > 0, P = P^T > 0$ e $Z = Z^T > 0$, matrizes de ponderação. $\tilde{\Theta} = \Theta - \Theta^*$ denota o erro de estimativa dos parâmetros neurais. O termo u_s em (3.62) é o controle CEV, usado para eliminar o efeito do erro de aproximação. Veja que em (3.62) o

termo $\Xi\Theta$ é responsável pela estimativa de $F_0(x_e)$ e $\Delta F_0(x_e)$. De maneira similar às soluções da matriz P apresentadas nos controladores robustos fuzzy descritos na seção anterior, a solução deste controlador é dada pela seguinte equação de Riccati

$$\dot{P} + PA_T + A_T^T P + PB_T \left(R^{-1} - \frac{1}{\gamma^2} I \right) B_T^T P + Q = 0 \quad (3.67)$$

que por sua vez pode ser simplificada para uma equação algébrica

$$\begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} - T_0^T [I \ 0]^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) [I \ 0] T_0 + Q = 0 \quad (3.68)$$

sendo P redefinida como

$$P = T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0.$$

Considerando o controlador $\mathcal{H}_\infty$ não linear baseado no modelo matemático do robô, que representa $F_0(x_e)$, e em redes neurais, para estimar o termo $\Delta F_0(x_e)$, vale a seguinte formulação: dado um nível de atenuação γ , encontre um controlador adaptativo baseado em modelo e em redes neurais com realimentação de estado

$$\tau = F_0(x_e) + \Xi\Theta + T_{11}^{-1}u + u_s \quad (3.69)$$

com

$$\dot{\Theta} = Proj[-Z^{-1}\Xi^T T_{11}[I \ 0]T_0\tilde{x}] \quad (3.70)$$

$$u = -R^{-1}[I \ 0]T_0\tilde{x} \quad (3.71)$$

$$u_s = -k(x_e)sgn(T_{11}[I \ 0]T_0\tilde{x}) \quad (3.72)$$

tal que o seguinte índice de desempenho seja satisfeito

$$\int_0^T (\tilde{x}^T Q \tilde{x} + u^T R u) dt \leq \tilde{x}^T(0)P\tilde{x}(0) + \tilde{\Theta}^T(0)Z\tilde{\Theta}(0) + \gamma^2 \int_0^T (d^T d)dt, \quad (3.73)$$

para matrizes de ponderação $Q = Q^T > 0$, $R = R^T > 0$, $P = P^T > 0$, $Z = Z^T > 0$ e $\tilde{\Theta} = \Theta - \Theta^*$ denota o erro de estimativa dos parâmetros neurais. O termo u_s em (3.69) é o controle por estrutura variável, usado para eliminar o efeito do erro de aproximação. Veja em (3.69), que o termo $\Xi\Theta$ para essa nova versão do controlador estima toda incerteza não modelada por $F_0(x_e)$.

De maneira similar aos controladores definidos anteriormente, P é uma matriz simétrica definida positiva cuja solução é dada pela seguinte equação de Riccati

$$\dot{P} + PA_T + A_T^T P + PB_T \left(R^{-1} - \frac{1}{\gamma^2} I \right) B_T^T P + Q = 0. \quad (3.74)$$

com as simplificações algébricas equivalentes

$$\begin{bmatrix} 0 & K \\ K & 0 \end{bmatrix} - T_0^T [I \ 0]^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) [I \ 0] T_0 + Q = 0. \quad (3.75)$$

sendo P redefinida como

$$P = T_0^T \begin{bmatrix} M_0 & 0 \\ 0 & K \end{bmatrix} T_0.$$

3.6 Solução da equação algébrica de Riccati

A equação algébrica de Riccati (3.26) pode ser resolvida com

$$T_0 = \begin{bmatrix} R_1^T Q_1 & R_1^T Q_2 \\ 0 & I \end{bmatrix},$$

$$K = \frac{1}{2} (Q_1^T Q_2 - Q_2^T Q_1) - \frac{1}{2} (Q_{21}^T + Q_{12}),$$

sendo a matriz R_1 o resultado da fatorização de Cholesky

$$R_1^T R_1 = \left(R^{-1} - \frac{1}{\gamma^2} I \right)^{-1},$$

a matriz simétrica definida positiva Q é fatorada como

$$Q = \begin{bmatrix} Q_1^T Q_1 & Q_{12} \\ Q_{12}^T & Q_2^T Q_2 \end{bmatrix},$$

e (3.26) fornece quatro equações distintas

$$Q_1^T Q_1 - T_{11}^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) T_{11} = 0, \quad (3.76)$$

$$K + Q_{12}^T - T_{12}^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) T_{11} = 0, \quad (3.77)$$

$$K + Q_{12} - T_{11}^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) T_{12} = 0, \quad (3.78)$$

$$Q_2^T Q_2 - T_{12}^T \left(R^{-1} - \frac{1}{\gamma^2} I \right) T_{12} = 0. \quad (3.79)$$

Capítulo 4

Implementação

4.1 Robô móvel com rodas

O RMR mostrado na Fig. 4.1 foi desenvolvido pelo Laboratório de Sistemas Inteligentes (LASI) da Universidade de São Paulo em São Carlos. Este RMR consiste de uma base de sustentação, dois motores DC com torque 0.29 N.m acoplados a rodas de borracha, seis pilhas recarregáveis que totalizam 7,2V e uma placa controladora.

A placa controladora consiste de um microcontrolador, um módulo RF (Rádio Frequência) e um módulo de acionamento dos motores.

O microcontrolador escolhido foi o MSP430F169 da Texas que possui as seguintes características: baixíssimo consumo de energia (330 μA no modo Active a 1 MHz e 2,2V; 1,1 μA no modo Standby; 0,2 μA no modo Off de retenção da RAM); alimentação entre 1,8 V a 3,6 V; CPU de 8 MHz com arquitetura RISC de 16 bits; 2048 B de RAM e 60 kB de memória de programa tipo flash, que pode ser usada para armazenar dados permanentes; 8 canais de conversor AD de 12 bits; conversor DA de 12 bits; 2 timers de 16 bits com gerador de PWM; 48 terminais (pinos) de entrada/saída (I/O); 2 interfaces de comunicação serial USART (SPI ou I2C); DMA; encapsulamento QFP com 64 terminais (pinos). A programação do MSP430F169 é feita no próprio circuito da placa controladora do robô diretamente de um computador, através de uma interface padrão JTAG.

O módulo de RF utiliza transceiver DR3000-1 da marca RF Monolithics (RFM). Foi desenvolvido também uma placa de interface da serial do computador com o transceiver DR3000-1, desse modo possibilitando a comunicação via rádio do computador com o RMR.

Figura 4.1: Foto do robô móvel.

O módulo de acionamento dos motores, consiste de um circuito integrado L293D, também da Texas que controla dois motores nos dois sentidos. O L293D trata-se de um circuito integrado contendo 4 meias ponte H, ou seja, 2 pontes H completas (2 canais). Possui capacidade máxima de corrente de saída de 600 mA por canal e opera cargas com tensão entre 4,5 V e 36 V.

Os parâmetros nominais do RMR da Fig. 4.1 são $a = 0.17(m)$, $b = 0.065(m)$, $d = 0.01(m)$, $r = 0.028(m)$, $m_r = 0.075(kg)$, $m_p = 0.597(kg)$ e momentos de inercias $I_c = 0.0022938(kg.m^2)$, $I_r = 0.000375(kg.m^2)$ e $I_m = 3.6788 \times 10^{-7}(kg.m^2)$.

4.2 Sistema de visão computacional

A localização do robô é feita encontrando a posição de centros de círculos em etiquetas fixadas sobre o robô em imagens bi-dimensionais monocromáticas, como mostrado em dos Reis (2005), que serão adquiridas através de um sistema de visão computacional que consiste de uma camera COHU H10x8M-II com resolução de 320 x 240 pixels e uma placa de aquisição PXC200F com taxa de processamento de 30 quadros/seg. Para o método de detecção de círculos da Seção 4.2.1, define-se a convenção utilizada para processamento de imagens, como mostrado na Fig. 4.2, sendo L o número de linhas e C o número de colunas da imagem digital, $f(i, j)$ a intensidade do nível de cinza da imagem na coordenada (i, j) com $f(i, j) \in \{0, 1, 2, 3, 4, \dots, 254, 255\}$, sendo o valor 0 representando a cor preta e 255 a branca.

Figura 4.2: Convenção: (a) imagem contínua, (b) imagem digital.

4.2.1 Transformada de Hough circular

Esta seção mostra um método para encontrar o centro de circunferências de raio fixo, R . Para isto utiliza-se uma máscara que também é uma circunferência de raio R , cuja função é atualizar os valores de uma matriz de dimensão $(L + 2R) \times (C + 2R)$ definida como matriz acumuladora.

O método Transformada de Hough Circular, mostrado em Sonka et al. (1998), consiste em posicionar o centro da máscara nos *pixels* da circunferência da imagem e os valores dos *pixels* nas posições correspondentes à circunferência da máscara na matriz acumuladora (inicialmente zerada) são incrementadas de uma unidade. A Fig. 4.3b ilustra bem o método,

Figura 4.3: Transformada de Hough circular: (a) imagem, (b) matriz acumuladora.

sendo as circunferências de linhas contínuas a contribuição da máscara, os losangos em negrito os *pixels* da circunferência da imagem considerada e o círculo em negrito a região de maior contribuição.

Após percorrer toda a imagem a matriz acumuladora estará totalmente atualizada, e, portanto, a posição do *pixel* de maior valor, denotada por $f_{max}(i, j)$, será considerada o centro da circunferência na matriz acumuladora e conseqüentemente o centro da circunferência na imagem será dado por

$$\bar{i} = i - R$$

e

$$\bar{j} = j - R.$$

O algoritmo 4.1 mostra como encontrar a postura de um robô usando a THC.

Algoritmo 4.1 Seqüência de passos para encontrar centros de circunferências em imagem usando a THC e determinar a postura de um RMR

- 1 - Coloque sobre o robô três etiquetas de circunferências de raio R (cor branca, por exemplo, para um campo de atuação de cor escura), tais que a média das duas posições dos centros das duas circunferências, localizadas no fundo do robô, seja o ponto P_o e a outra circunferência localizada no eixo de simetria, na frente do robô, bem afastada das outras duas;
 - 2 - Calibre o sistema capturando uma imagem e verificando o menor valor dos *pixels* pertencentes às circunferências;
 - 3 - A cada quadro de imagem capturada, verificar quais os *pixels* são supostamente pertencentes às circunferências (Valor do nível de cinza maior ou igual ao menor valor encontrado na calibração) e atualizar a matriz acumuladora, conforme apresentada anteriormente;
 - 4 - Encontre a posição na matriz acumuladora de maior valor de nível de cinza $f_{max}(i, j)$.
E, portanto, a posição da primeira circunferência na imagem será $(\bar{i}_1, \bar{j}_1) = (i - R, j - R)$;
 - 5 - Zerar uma região na matriz acumuladora nas proximidades de $(i - R, j - R)$;
 - 6 - Encontre a posição na matriz acumuladora de maior valor de nível de cinza $f_{max}(i, j)$.
E, portanto, a posição da segunda circunferência na imagem será $(\bar{i}_2, \bar{j}_2) = (i - R, j - R)$;
 - 7 - Zerar uma região na matriz acumuladora nas proximidades de $(i - R, j - R)$;
 - 8 - Encontre a posição na matriz acumuladora de maior valor de nível de cinza $f_{max}(i, j)$.
E, portanto, a posição da terceira circunferência na imagem será $(\bar{i}_3, \bar{j}_3) = (i - R, j - R)$;
 - 9 - A posição do robô, $(x_o, y_o) = (\bar{i}_o, \bar{j}_o)$, será a média entre as posições dos dois centros das circunferências mais próximas;
 - 10- A direção do robô será dada pela inclinação da reta que passa por $(\bar{i}_o, \bar{j}_o)$ e a posição do centro da circunferência mais distante.
-

4.3 Ambiente de controle de RMR

A interface entre as técnicas de controle desenvolvidas no Capítulo 3 e o robô móvel com rodas é feita através do ambiente de controle de RMR (ACRMR), que foi desenvolvido em C++. O ACRMR funciona da seguinte maneira: a aquisição da imagem através da câmera e localização do robô utilizando THC é feita em paralelo ao controle do RMR, que gera os torques que são transmitidos pela porta serial do computador para a placa de interface do computador com o rádio. O sistema de controle de acompanhamento de trajetória para RMR pode ser melhor entendido na Seção 4.4.

O ACRMR possibilita que o usuário visualize a imagem da câmera em uma janela a parte Fig. 4.3. Escolha qual controlador deseja utilizar através da caixa de seleção do controlador. Há a possibilidade de alterar os parâmetros nominais do RMR e os ganhos dos controladores na aba *Parâmetros* Fig. 4.5. Visualize os gráficos referentes ao experimento nas abas *Gráficos A* Fig. 4.6 e *Gráficos B* Fig. 4.7. A visualização dos dados de postura do RMR pode ser obtida na tela principal. O controle manual do robô pode ser feito via teclado e, através do menu arquivo, é possível configurar a porta serial do microcomputador e exportar os dados dos experimentos para arquivos que podem ser lidos pelo *Matlab*.

Figura 4.4: Imagem da câmera visualizando o robô.

Figura 4.5: Ambiente de controle de RMR na aba de Parâmetros.

Figura 4.6: Ambiente de controle de RMR na aba de Gráfico A.

Figura 4.7: Ambiente de controle de RMR na aba de Gráfico B.

4.4 Experimentos

Os resultados experimentais foram obtidos usando o RMR descrito na Seção 4.1, o sistema de visão da Seção 4.2 e o ACRMR da Seção 4.3. A taxa de amostragem foi ajustada para 33 ms, a qual é utilizada para aquisição de imagem do sistema de visão computacional. Os comandos de controle são gerados pelo ACRMR instalado em um computador com processador Pentium 4 de 3.2 GHz e 2GB de memória RAM e são enviados para o RMR através de um módulo de rádio frequência com taxa de transmissão de 38400 bps.

A trajetória de referência é definida por $x_r = 0.0339t$ e $y_r = 0.1132\text{sen}(0.3t)$, com a condição inicial $(x_{o_o}, y_{o_o}, \alpha_o) = (0, 0, 0)$. Para o controlador baseado na cinemática, os ganhos foram selecionados como $k_x = 0.75$, $k_y = 200$ e $k_\alpha = 25.5$. Para permitir uma análise de robustez dos controladores, distúrbios externos (Fig.4.8) foram introduzidos nos torques das rodas da forma

$$w_d = 0.05e^{-(t-6)^4}\text{sen}(1.3\pi t) \quad \text{e} \quad w_e = -0.15e^{-(t-6)^4}\text{sen}(1.3\pi t).$$

Figura 4.8: Distúrbios de torque aplicados às rodas.

A Fig. 4.9 mostra o diagrama de blocos completo da estratégia de controle do robô móvel. E a Fig. 4.10 mostra os controladores baseados na dinâmica. A Fig. 4.10 (a) mostra os controladores baseados somente no modelo do RMR, a Fig. 4.10 (b) o controlador robusto baseado em modelo fuzzy T-S, a Fig. 4.10 (c) o controlador robusto baseado em modelo e em modelo fuzzy T-S, a Fig. 4.10 (d) o controlador robusto baseado em redes neurais e a Fig. 4.10 (e) o controlador robusto baseado em modelo e em redes neurais. O controlador Proporcional Derivativo em conjunto com Torque Calculado (PD + TC) será apresentado na Seção 4.4.7.

Figura 4.9: Sistema de controle de acompanhamento de trajetória para robôs móveis.

Figura 4.10: Controladores baseados na dinâmica

4.4.1 Controle $\mathcal{H}_\infty$ não linear via representação quase-LPV

A matriz do sistema $A(\dot{q}_2)$ está em função apenas das velocidades das rodas $\dot{q}_2$. Portanto, no projeto do controlador quase-LPV pode-se considerar como função dos erros de velocidades $\dot{\tilde{q}}_2$. Assim o parâmetro ρ é definido como $\rho(\tilde{x}) = \dot{\tilde{q}}_2$. O conjunto P é dado por

$$-\pi \leq \dot{\tilde{\theta}}_d \leq \pi(\text{rad/s}) \text{ e } -\pi \leq \dot{\tilde{\theta}}_e \leq \pi(\text{rad/s}),$$

sendo $\nu = [\ddot{\tilde{\theta}}_{d_{max}} \ \ddot{\tilde{\theta}}_{e_{max}}] = [2.5\pi \ 2.5\pi](\text{rad/s}^2)$, e P dividido em $N = 3$ para cada parâmetro $\dot{\tilde{\theta}}_d$ e $\dot{\tilde{\theta}}_e$. As funções base, que são dependentes dos parâmetros, são selecionadas da seguinte forma

$$\begin{aligned} f_1 &= 10 + \cos \dot{\tilde{\theta}}_d + \sin \dot{\tilde{\theta}}_d + \cos \dot{\tilde{\theta}}_e + \sin \dot{\tilde{\theta}}_e, \\ f_2 &= \cos \dot{\tilde{\theta}}_d + \sin \dot{\tilde{\theta}}_e \text{ e} \\ f_3 &= -\sin \dot{\tilde{\theta}}_d + \cos \dot{\tilde{\theta}}_e. \end{aligned}$$

Essas funções foram selecionadas de maneira a acentuar a influência da variação do erro de velocidade das rodas no cálculo dos ganhos dos torques. A desigualdade (3.18) é resolvida usando o software MATLAB[®] [Gahinet et al. (1995)], sendo que em (3.16)

$$x = \tilde{x}, \quad A(\rho(t)) = \begin{bmatrix} A(\dot{\tilde{q}}_2) & 0 \\ I & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} I_{2 \times 2} \\ 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} B \\ 0 \end{bmatrix},$$

$$C_1 = I_{4 \times 4} \text{ e } C_2 = 0.$$

As matrizes X_i são encontradas para o melhor nível de atenuação $\gamma = 1655$ (testes realizados com as perturbações e incertezas paramétricas) e são dadas por

$$X_1 = 10^{-2} \times \begin{bmatrix} 1.213 & 0.132 & -1.837 & -0.109 \\ 0.132 & 1.213 & -0.109 & -1.837 \\ -1.837 & -0.109 & 3.040 & 0.096 \\ -0.109 & -1.837 & 0.096 & 3.040 \end{bmatrix},$$

$$X_2 = 10^{-2} \times \begin{bmatrix} -1.214 & -0.132 & 1.841 & 0.109 \\ -0.132 & -1.214 & 0.109 & 1.841 \\ 1.841 & 0.109 & -3.044 & -0.097 \\ 0.109 & 1.841 & -0.097 & -3.044 \end{bmatrix},$$

$$X_3 = 10^{-2} \times \begin{bmatrix} 1.195 & 0.127 & -1.781 & -0.099 \\ 0.127 & 1.195 & -0.099 & -1.781 \\ -1.781 & -0.099 & 2.978 & 0.090 \\ -0.099 & -1.781 & 0.090 & 2.978 \end{bmatrix}.$$

Os resultados são mostrados nas Figs. 4.11 a 4.18.

Figura 4.11: Acompanhamento de trajetória de referência usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.12: Erros de posição usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.13: Erro de direção usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.14: Derivada dos erros de posição usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.15: Derivada do erro de direção usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.16: Velocidade angular da roda direita usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.17: Velocidade angular da roda esquerda usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.18: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador quase-LPV: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.2 Controle $\mathcal{H}_\infty$ não linear via Teoria dos Jogos (TJ)

O controlador baseado na TJ é caracterizado por um nível de atenuação $\gamma = 66.632$. As matrizes de ponderação selecionadas são $Q_1 = I_{(2 \times 2)}$, $Q_2 = 0.5 \times I_{(2 \times 2)}$, $Q_{12} = 0$ e $R = 10.8 \times I_{(2 \times 2)}$. Os resultados são mostrados nas Figs. 4.19 e 4.26.

Figura 4.19: Acompanhamento de trajetória de referência usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.20: Erros de posição usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.21: Erro de direção usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.22: Derivada dos erros de posição usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.23: Derivada do erro de direção usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.24: Velocidade angular da roda direita usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.25: Velocidade angular da roda esquerda usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.26: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador TJ: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.3 Controle $\mathcal{H}_\infty$ não linear baseado em modelo fuzzy Takagi Sugeno

Um conjunto de sistemas fuzzy é definido

$$Y(\tilde{x}, A(\tilde{x}), \Theta) := \begin{bmatrix} y^1([\tilde{q}(1) \ \dot{\tilde{q}}(1)], A(\tilde{x}), \Theta_1) \\ y^2([\tilde{q}(2) \ \dot{\tilde{q}}(2)], A(\tilde{x}), \Theta_2) \end{bmatrix}$$

sendo $y^1(\cdot)$ e $y^2(\cdot)$ a estimativa da dinâmica das rodas direita e esquerda, respectivamente. Os conjuntos fuzzy $A(\tilde{x})$ são definidos para o universo de discurso de erro de deslocamento, $u_1 = \tilde{q} \in U_1$, e para o universo de discurso de erro de velocidade, $u_2 = \dot{\tilde{q}} \in U_2$, na Fig. 4.27.

Figura 4.27: Conjunto fuzzy $A_1(\tilde{q})$ e conjunto fuzzy $A_2(\dot{\tilde{q}})$

A base de regra fuzzy é dada por

- R_1 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_1
- R_2 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Zero)) **ENTÃO** y_2
- R_3 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_3
- R_4 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_4
- R_5 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Zero)) **ENTÃO** y_5
- R_6 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_6
- R_7 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_7
- R_8 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Zero)) **ENTÃO** y_8
- R_9 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_9

O controlador $\mathcal{H}_\infty$ é caracterizado por um nível de atenuação $\gamma = 66.632$ e $k(x_e) = \frac{1}{100}\sqrt{\tilde{x}^T \tilde{x}}$.

As matrizes de ponderação selecionadas são $Q_1 = I_{(2 \times 2)}$, $Q_2 = 0.5 \times I_{(2 \times 2)}$, $Q_{12} = 0$ e $R = 10.8 \times I_{(2 \times 2)}$ e $H = 2000 \times I_{(54 \times 54)}$. A lei de controle adaptativa para ajustar y_i é implementada baseada em (3.35).

Os resultados são mostrados nas Figs. 4.28 a 4.35.

Figura 4.28: Acompanhamento de trajetória de referência usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.29: Erros de posição usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.30: Erro de direção usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.31: Derivada dos erros de posição usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.32: Derivada do erro de direção usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.33: Velocidades angulares da roda direita usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.34: Velocidades angulares da roda esquerda usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.35: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.4 Controle $\mathcal{H}_\infty$ não linear baseado em modelo e em modelo fuzzy Takagi Sugeno

Um conjunto de sistemas fuzzy é definido como

$$Y(\tilde{x}, A(\tilde{x}), \Theta) := \begin{bmatrix} y^1([\tilde{q}(1) \ \dot{\tilde{q}}(1)], A(\tilde{x}), \Theta_1) \\ y^2([\tilde{q}(2) \ \dot{\tilde{q}}(2)], A(\tilde{x}), \Theta_2) \end{bmatrix}$$

sendo $y^1(\cdot)$ e $y^2(\cdot)$ a estimativa da dinâmica da roda direita e esquerda, respectivamente. Os conjuntos fuzzy $A(\tilde{x})$ são definidos para o universo de discurso de erro de deslocamento, $u_1 = \tilde{q} \in U_1$, e para o universo de discurso de erro de velocidades, $u_2 = \dot{\tilde{q}} \in U_2$, na Fig. 4.36.

Figura 4.36: Conjunto fuzzy $A_1(\tilde{q})$ e conjunto fuzzy $A_2(\dot{\tilde{q}})$

A base de regra fuzzy é dada por

- R_1 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_1
- R_2 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Zero)) **ENTÃO** y_2
- R_3 : **SE** (u_1 é A_1 (Negativo)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_3
- R_4 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_4
- R_5 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Zero)) **ENTÃO** y_5
- R_6 : **SE** (u_1 é A_1 (Zero)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_6
- R_7 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Negativo)) **ENTÃO** y_7
- R_8 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Zero)) **ENTÃO** y_8
- R_9 : **SE** (u_1 é A_1 (Positivo)) e (u_2 é A_2 (Positivo)) **ENTÃO** y_9

O controlador $\mathcal{H}_\infty$ é caracterizado por um nível de atenuação $\gamma = 66.632$ e $k(x_e) = \frac{1}{100} \sqrt{\tilde{x}^T \tilde{x}}$. As matrizes de ponderação selecionadas são $Q_1 = I_{(2 \times 2)}$, $Q_2 = 0.5 \times I_{(2 \times 2)}$, $Q_{12} = 0$ e $R = 10.8 \times I_{(2 \times 2)}$ e $H = 2000 \times I_{(54 \times 54)}$. A lei de controle adaptativa para ajustar y_i é implementada baseada em (3.35). Os resultados são mostrados nas figs. 4.37 a 4.44.

Figura 4.37: Acompanhamento de trajetória de referência usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.38: Erros de posição usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.39: Erro de direção usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.40: Derivada dos erros de posição usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.41: Derivada do erro de direção usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.42: Velocidades angulares da roda direita usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.43: Velocidades angulares da roda esquerda usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.44: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo e em modelo fuzzy T-S: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.5 Controle $\mathcal{H}_\infty$ não linear baseado em redes neurais

O controlador $\mathcal{H}_\infty$ é caracterizado por um nível de atenuação $\gamma = 66.632$ e $k(x_e) = \frac{1}{200}\sqrt{\tilde{x}^T \tilde{x}}$. As matrizes de ponderação selecionadas foram $Q_1 = I_{(2 \times 2)}$, $Q_2 = 0.5 \times I_{(2 \times 2)}$, $Q_{12} = 0$, $R = 10.8 \times I_{(2 \times 2)}$ e $Z = 3000 \times I_{(14 \times 14)}$. A lei de controle adaptativa para ajustar y_i é implementada baseada na Equação (3.63). Definem-se $\hat{F}(x_e, \Theta) = \begin{bmatrix} \hat{F}_1(x_e, \Theta_1) & \hat{F}_2(x_e, \Theta_2) \end{bmatrix}^T$ com $p_k = 7$ neurônios na camada escondida, o vetor de bias $b_k = [-3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3]$ e a matriz de pesos para a primeira camada $\Omega_i^k = [\omega_{ij}^k] = [-1 \quad -1 \quad -1 \quad -1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]$. Os parâmetros Θ são definidos como

$$\Theta = \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix}$$

com

$$\begin{aligned} \Theta_1 &= [\theta_{11} \quad \theta_{12} \quad \theta_{13} \quad \theta_{14} \quad \theta_{15} \quad \theta_{16} \quad \theta_{17}]^T, \\ \Theta_2 &= [\theta_{21} \quad \theta_{22} \quad \theta_{23} \quad \theta_{24} \quad \theta_{25} \quad \theta_{26} \quad \theta_{27}]^T \end{aligned}$$

e a matriz Ξ pode ser calculada como

$$\Xi = \begin{bmatrix} \xi_1^T & 0 \\ 0 & \xi_2^T \end{bmatrix},$$

com $\xi_1^T = [\xi_{11}, \dots, \xi_{17}]$ e $\xi_2^T = [\xi_{21}, \dots, \xi_{27}]$.

Os resultados são mostrados nas Figs. 4.45 a 4.52.

Figura 4.45: Acompanhamento de trajetória de referência usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.46: Erros de posição usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.47: Erro de direção usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.48: Derivada dos erros de posição usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.49: Derivada do erro de direção usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.50: Velocidades angulares da roda direita usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.51: Velocidades angulares da roda esquerda usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.52: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.6 Controle $\mathcal{H}_\infty$ não linear baseado em modelo e em redes neurais

O controlador $\mathcal{H}_\infty$ é caracterizado por um nível de atenuação $\gamma = 66.632$ e $k(x_e) = \frac{1}{200}\sqrt{\tilde{x}^T \tilde{x}}$. As matrizes de ponderação selecionadas foram $Q_1 = I_{(2 \times 2)}$, $Q_2 = 0.5 \times I_{(2 \times 2)}$, $Q_{12} = 0$, $R = 10.8 \times I_{(2 \times 2)}$ e $Z = 3000 \times I_{(14 \times 14)}$.

A lei de controle adaptativa para ajustar y_i é implementada baseada na Equação (3.70). Definem-se $\Delta \hat{F}(x_e, \Theta) = \begin{bmatrix} \Delta \hat{F}_1(x_e, \Theta_1) & \Delta \hat{F}_2(x_e, \Theta_2) \end{bmatrix}^T$ com $p_k = 7$ neurônios na camada escondida, o vetor de bias $b_k = \begin{bmatrix} -3 & -2 & -1 & 0 & 1 & 2 & 3 \end{bmatrix}$ e a matriz de pesos para a primeira camada $\Omega_i^k = [\omega_{ij}^k] = \begin{bmatrix} -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$. Os parâmetros Θ são definidos como

$$\Theta = \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix}$$

com

$$\begin{aligned} \Theta_1 &= \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} & \theta_{14} & \theta_{15} & \theta_{16} & \theta_{17} \end{bmatrix}^T, \\ \Theta_2 &= \begin{bmatrix} \theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} & \theta_{25} & \theta_{26} & \theta_{27} \end{bmatrix}^T \end{aligned}$$

e a matriz Ξ pode ser calculada como

$$\Xi = \begin{bmatrix} \xi_1^T & 0 \\ 0 & \xi_2^T \end{bmatrix},$$

com $\xi_1^T = [\xi_{11}, \dots, \xi_{17}]$ e $\xi_2^T = [\xi_{21}, \dots, \xi_{27}]$.

Os resultados são mostrados nas Figs. 4.53 a 4.60.

Figura 4.53: Acompanhamento de trajetória de referência usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.54: Erros de posição usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.55: Erro de direção usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.56: Derivada dos erros de posição usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.57: Derivada do erro de direção usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.58: Velocidades angulares da roda direita usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.59: Velocidades angulares da roda esquerda usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.60: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador baseado em modelo e em redes neurais: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.7 Controle proporcional derivativo e torque calculado (PD+TC)

O controle PD consiste de um controle independente para cada roda, dado por

$$\begin{aligned} u_d &= -[\theta_d - \theta_d^d]k_{p1} - [\dot{\theta}_d - \dot{\theta}_d^d]k_{d1}, \\ u_e &= -[\theta_e - \theta_e^d]k_{p2} - [\dot{\theta}_e - \dot{\theta}_e^d]k_{d2}, \end{aligned} \quad (4.1)$$

sendo os ganhos proporcionais e derivativos ajustados em $K_{p1} = K_{p2} = 50$ e $K_{d1} = K_{d2} = 100$, tais que os erros de postura sem distúrbios e incertezas fiquem próximos aos dos erros dos controladores via controle $\mathcal{H}_\infty$. Os torques aplicados ao robô móvel são obtidos considerando (3.5), que de fato é a equação conhecida do torque calculado. Os resultados são mostrados nas figs. 4.61 a 4.68.

Figura 4.61: Acompanhamento de trajetória de referência usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.62: Erros de posição usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.63: Erro de direção usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.64: Derivada dos erros de posição usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.65: Derivada do erro de direção usando o controlador PD: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.66: Velocidades angulares da roda direita usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.67: Velocidades angulares da roda esquerda usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

Figura 4.68: Torques τ_d e τ_e aplicados às rodas direita e esquerda, respectivamente, usando o controlador PD + TC: sem distúrbio (esquerda) e com distúrbio (direita).

4.4.8 Estudo comparativo

Os controladores foram comparados baseados nos seguintes índices de desempenho

- Norma $\mathcal{L}_2$ dos erros de posição e velocidade

$$\mathcal{L}_2[e] = \left(\frac{1}{(t_r - t_0)} \int_{t_0}^{t_r} \|e\|_2^2 dt \right)^{\frac{1}{2}},$$

sendo $e = \begin{bmatrix} q_e & \dot{q}_e \end{bmatrix}^T$,

- Energia total

$$E[\tau] = \sum_{i=1}^2 \left(\int_{t_0}^{t_r} \|\tau_i\| dt \right)$$

sendo $t_o = 0$ até $t_r = 20(s)$ o tempo de experimento e $\tau_i(t)$ o torque da roda i (direita ou esquerda).

Os resultados estão apresentados na Tabela 4.1.

Tabela 4.1: Índices de desempenho.

Controlador	Sem distúrbios		Com distúrbios	
	$\mathcal{L}_2[e]$	$E[\tau]$	$\mathcal{L}_2[e]$	$E[\tau]$
$\mathcal{H}_\infty$ Não linear via quase-LPV	0.1585	2.5443	0.1695	2.6500
$\mathcal{H}_\infty$ Não linear via TJ	0.1522	2.5695	0.1607	2.6619
Adap. $\mathcal{H}_\infty$ Não linear T-S	0.1523	2.6067	0.1642	2.6797
Adap. $\mathcal{H}_\infty$ Não linear modelo + T-S	0.1543	2.6019	0.1588	2.6168
Adap. $\mathcal{H}_\infty$ Não linear RN	0.1519	2.6818	0.1649	2.6903
Adap. $\mathcal{H}_\infty$ Não linear modelo + RN	0.1524	2.5795	0.1596	2.6215
PD+TC	0.1559	2.5173	0.2833	3.2083

Observa-se na Tabela 4.1 que sem a presença de distúrbios, os controladores robustos e o controlador PD + TC apresentaram resultados equivalentes do ponto de vista do erro acompanhamento de trajetória. Vale ressaltar que o controlador PD + TC apresentou melhor economia de energia. Com a presença de distúrbios verifica-se que os controladores robustos apresentaram melhores resultados, em ambos os índices utilizados, que o controlador PD + TC. Os controladores robustos baseados em modelo e em técnicas inteligentes apresentaram melhores resultados tanto com relação ao erro de acompanhamento de trajetória como na economia de energia em relação aos demais controladores robustos. A abordagem utilizando modelo fuzzy

Takagi Sugeno se destacou com relação a abordagem com redes neurais, nesses experimentos. Os números obtidos na Tabela 4.1 estão baseados na média de cinco experimentos realizados com o RMR. Para generalizar uma possível superioridade dos modelos fuzzy com relação às redes neurais para este tipo de problema, há certamente a necessidade de mais estudos. Esse é um dos aspectos a serem explorados em trabalhos futuros.

Capítulo 5

Conclusão

Seis controladores robustos baseados no modelo dinâmico de um RMR e em técnicas inteligentes, utilizando técnicas de controle $\mathcal{H}_\infty$ foram propostos para o problema de acompanhamento de trajetória. Um ambiente de controle de RMRs (ACRMR) foi desenvolvido para realização de experimentos e estudos com RMRs. Um estudo comparativo entre os controladores considerando distúrbios externos foi realizado. Resultados experimentais têm mostrado que é bastante eficiente utilizar técnicas inteligentes para estimar apenas a parte incerta do modelo nominal em sistemas de controle robustos. Principalmente se considerarmos que nas estratégias adotadas para esses controladores há garantia de estabilidade do sistema e o critério $\mathcal{H}_\infty$ é satisfeito.

Referências Bibliográficas

- Aicardi, M., G. Casalino, A. Bicchi, e A. Balestrino (1995). Closed loop steering of unicycle like vehicles via lyapunov techniques. *IEEE Robotics & Automation Magazine* 2(1), 27–35.
- Almeida, J., F. L. Pereira, e J. B. Sousa (1997). A hybrid feedback control system for a nonholonomic car-like vehicle. In *IEEE International Conference on Robotics and Automation*, Volume 3, pp. 2614 – 2619.
- Apkarian, P. e R. J. Adams (1998). Advanced gain-scheduling techniques for uncertain systems. *IEEE Transactions on Control Systems Technology* 6(1), 21–32.
- Basar, T. e P. Bernhard (1990). *$\mathcal{H}_\infty$ -Optimal Control and Related Minimax Problems*. Berlin: Birkhauser.
- Basar, T. e J. Olsder (1982). *Dynamic Noncooperative Game Theory*. New York: Academic Press.
- Bianchi, R. A. C., A. S. Simões, e A. H. R. Costa (2001). Comportamentos reativos para seguir pistas em um robô móvel guiado por visão. *Simpósio Brasileiro de Automação Inteligente* (5, Canela).
- Bloch, A. M. e N. H. McClamroch (1989). Control of mechanical systems with classical nonholonomic constraints. In *IEEE Conference on Decision and Control*, Volume 1, pp. 201–205.
- Bloch, A. M. e N. H. McClamroch (1990). Controllability and stabilizability properties of a nonholonomic control system. In *IEEE Conference on Decision and Control*, Volume 3, pp. 1312 – 1314.
- Bozma, O. e R. Kuc (1994). A physical model-based analysis of heterogeneous environments using sonar-endura method. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(5), 497 – 506.

- Brockett, R. W. (1983). Asymptotic stability and feedback stabilization. In *Differential Geometric Control Theory*, R. W. Brockett, R. S. Millman and H. J. Sussmann, Boston, MA: Birkhäuser, pp. 181–191.
- Campion, G., G. Bastin, e B. D’Andréa-Novet (1996). Strutural properties and classification of kinematic and dynamic model of wheeled mobile robots. *IEEE Transactions on Robotics and Automation* 12(1), 47–62.
- Chang, Y. C. (2000). Neural network-based $\mathcal{H}_\infty$ tracking control for robotic systems. *IEE Proceedings of Control Theory Applications* 147(3), 303–311.
- Chang, Y. C. (2005). Intelligent robust control for uncertain nonlinear time-varying systems and its application to robotic systems. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics* 35(6), 1108–1119.
- Chang, Y. C. e B. S. Chen (1997). A nonlinear adaptive $\mathcal{H}_\infty$ tracking control design in robotic systems via neural networks. *IEEE Transactions on Control Systems Technology* 5(1), 13–29.
- Chen, B. S., Y. C. Chang, e T. C. Lee (1997). Adaptive control in robotic systems with $\mathcal{H}_\infty$ tracking performance. *Automatica* 33(2), 227–234.
- Chen, B. S., T. S. Lee, e J. H. Feng (1994). A nonlinear $\mathcal{H}_\infty$ control design in robotic systems under parameter perturbation and external disturbance. *International Journal of Control* 59(2), 439–461.
- Coelho, P. (2001). Aplicação de lagrangianos na dinâmica de robôs móveis com rodas. *Technical Report, No. ISR-LCIR-2001/01 (Instituto de sistemas e robótica), Universidade de Coimbra, Portugal*.
- Coelho, P. e U. Nunes (2003). Lie algebra application to mobile robot control: A tutorial. *Robotica* 21, 483 – 493.
- Corradini, M. L., G. Ippoliti, e S. Longhi (2003). Neural networks based control of mobile robots: Development and experimental validation. *Journal of Robotic Systems* 20(10), 587–600.
- Craig, J. (1985). *Adaptive Control of Mechanical Manipulators*. Addison-Wesley.
- Do, K. D., Z. P. Jiang, e J. Pan (2004). Simultaneous tracking and stabilization of mobile robots: an adaptive approach. *IEEE Transactions on Automatic Control* 49(7), 1147 – 1151.

- dos Reis, G. A. (2005). Controle $\mathcal{H}_\infty$ não linear de robôs móveis com rodas. Dissertação de Mestrado, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos. 2005.
- Fukao, T., H. Nakagawa, e N. Adachi (2000). Adaptive tracking control of a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation* 16(5), 609 – 615.
- Gahinet, P., A. Nemirovski, A. J. Laub, e M. Chilali (1995). *LMI Control Toolbox*. The MathWorks Inc.
- Ge, S. S., T. H. Lee, e C. J. Harris (1998). *Adaptive Neural Network Control of Robotic Manipulators*. Singapore: World Scientific.
- Gupta, G. S., C. H. Messom, e S. Demidenko (2005). Real-time identification and predictive control of fast mobile robots using global vision sensing. *IEEE Transactions on Instrumentation and Measurement* 54(1), 200 – 214.
- Ho, C. T. e L. H. Chen (1995). A fast ellipse/circle detector using geometric symmetry. *Pattern Recognition* 28(1), 117–124.
- Huang, Y. e A. Jadbabaie (1998). Nonlinear $\mathcal{H}_\infty$ control: An enhanced quasi-lpv approach. In *IEEE International Conference on Decision and Control*, 37, Tampa, Florida, USA. Workshop in $\mathcal{H}_\infty$ nonlinear control by J. C. Doyle, Caltech.
- Hwang, C. K., B. S. Chen, e Y. T. Chang (2004). Combination of kinematical and robust dynamical controllers for mobile robotics tracking control: (i) optimal $\mathcal{H}_\infty$ control. In *Proceedings IEEE International Conference on Control Applications*, pp. 1205–1210.
- Ji, M., Z. Zhang, G. Biswas, e N. Sarkar (2003). Hybrid fault adaptive control of a wheeled mobile robot. *IEEE/ASME Transactions on Mechatronics* 8(2), 226 – 233.
- Jiang, Z.-P. e H. Nijmeijer (1999). A recursive technique for tracking control of nonholonomic systems in chained form. *IEEE Transactions on Automatic Control* 44(2), 265 – 279.
- Johansson, R. (1990). Quadratic optimization of motion coordination and control. *IEEE Transactions on Automatic Control* 35(11), 1197–1208.
- Kanayama, Y., Y. Kimura, F. Miyazaki, e T. Noguchi (1990). A stable tracking control method for an autonomous mobile robot. In *Proceedings IEEE International Conference on Robotics and Automation*, pp. 384 – 389.

- Kimme, C. e J. Sklansky (1975). Finding circles by an array of accumulators. *Communication of the ACM* 18(2), 120–122.
- Krstic, M., I. Kanellakopoulos, e P. Kokotovic (1995). *Nonlinear and Adaptive Control Design*. John Wiley and Sons, Inc., 605 Third Avenue, New York, NY.
- Lee, P., T. Dean, A. Yap, D. Walter, L. Kitchen, e N. Barnes (2004). On-board vision using visual-servoing for robocup f-180 league mobile robots. *ROBOCUP 2003: ROBOT SOCCER WORLD CUP VII LECTURE NOTES IN COMPUTER SCIENCE* 3020, 422–433.
- Lee, T.-C., K.-T. Song, C.-H. Lee, e C.-C. Teng (2001). Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. *IEEE Transactions on Control Systems Technology* 9(2), 305 – 318.
- Lewis, F. L., C. T. Abdallah, e D. M. Dawson (1993). *Control of robot manipulators*. New York: Macmillan.
- Liu, H., H. Zha, e F. Lin (2004). Competition analysis system for soccer robots based on global vision and trajectory restrictions. *IEEE International Conference on Systems, Man and Cybernetics* 6, 5296 – 5300.
- Luca, A. D., G. Oriolo, L. Paone, e P. R. Giordano (2002). Experiments in visual feedback control of a wheeled mobile robot. In *IEEE Conference on Robotics and Automation*, pp. 2073–2078.
- McLaughlin, R. A. e M. D. Alder (1998). The hough transform versus the upwrite. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(4), 396–400.
- M'Closkey, R. T. e R. M. Murray (1997). Exponential stabilization of driftless nonlinear control systems using homogeneous feedback. *Automatic Control, IEEE Transactions on* 42(5), 614 – 628.
- Morin, P. e C. Samson (2000). Practical stabilization of a class of nonlinear systems. application to chain systems and mobile robots. In *IEEE Conference on Decision and Control*, Volume 3, pp. 2989 – 2994.
- Oriolo, G., A. D. Luca, e M. Vendittelli (2002). Wmr control via dynamic feedback linearization: Design, implementation, and experimental validation. *IEEE Transactions Control Systems Technology* 10(6), 835–852.

-
- Postlethwaite, I. e A. Bartoszewicz (1998). Application of non-linear $\mathcal{H}_\infty$ control to the tetrabot robot manipulator. *Proceedings of the Institution of Mechanical Engineers - Part I - Journal of Systems and Control Engineering* 212(16), 459–465.
- Samson, C. (1995). Control of chained systems application to path following and time-varying point-stabilization of mobile robots. *IEEE Transactions on Automatic Control* 40(1), 64–77.
- Sarkar, N., X. Yun, e V. Kumar (1994). Control of mechanical systems with rolling constraints: Application to dynamic control of mobile robots. *Int. J. Rob. Res.* 13(1), 55–69.
- Siqueira, A. A. G. e M. H. Terra (2004). Nonlinear and markovian $\mathcal{H}_\infty$ controls of underactuated manipulators. *IEEE Transaction on Control Systems Technology* 12(6), 811–826.
- Sonka, M., V. Hlavac, e R. Boyle (1998). *Image Processing, Analysis, and Machine Vision*. Wiley Inter Science.
- Takagi, T. e M. Sugeno (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics* 15, 116–132.
- Vendittelli, M. e G. Oriolo (2000). Stabilization of the general two-trailer system. In *IEEE International Conference on Robotics and Automation*, Volume 2, pp. 1817 – 1823.
- Wu, F. (1995). *Control of linear parameter-varying systems*. 150p. PhD Thesis - Department of Mechanical Engineering, University of California, Berkeley. 1995.
- Wu, F., X. H. Yang, A. Packard, e G. Becker (1996). Induced $\mathcal{L}_2$ -norm control for lpv systems with bounded parameter variation rates. *International Journal of Robust and Nonlinear Control* 6(9-10), 983–998.
- Yagi, Y., S. Kawato, e S. Tsuji (1994). Real-time omnidirectional image sensor (copis) for vision-guided navigation. *IEEE Transactions on Robotics and Automation* 10(1), 11–22.