



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Controle Dinâmico de Robôs Móveis com Acionamento Diferencial

Autor: Frederico Carvalho Vieira
Orientador: Prof. Dr. Adelardo Adelino Dantas de Medeiros
Co-Orientador: Prof. Dr. Pablo Javier Alsina

Natal–RN Brasil
Fevereiro de 2005



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Controle Dinâmico de Robôs Móveis com Acionamento Diferencial

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Rio Grande do Norte como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência.

Natal–RN Brasil
Fevereiro de 2005

Banca examinadora:

Prof. Dr. Adelardo Adelino Dantas de Medeiros
Universidade Federal do Rio Grande do Norte

Prof. Dr. Carlos Eduardo Trabuco Dórea
Universidade Federal da Bahia

Prof. Dr. Pablo Javier Alsina
Universidade Federal do Rio Grande do Norte

Prof. Dr. Aldayr Dantas de Araújo
Universidade Federal do Rio Grande do Norte

Agradecimentos

À Deus por permitir receber mais esta benção em minha vida.

À minha filha Ana Clara pelo simples fato de poder me fazer sorrir e amá-la infinitamente em todos os dias de minha vida.

À minha esposa Maysa por toda dedicação, amor e compreensão que me fazem a cada dia mais feliz, e de onde busquei o incentivo para atingir mais esta conquista.

Ao meus pais José Costa e Vitória, pelo amor, educação, carinho, e pela insistência na solicitação de meu empenho para superar os obstáculos da vida. Além disso, eles são a base de todas as minhas conquistas.

À minha irmã Andréa por estar sempre presente em todos os momentos de minha vida, com muito carinho e incentivo.

Aos meus tios Lourdinha e Nepomuceno, por todo carinho e por terem me acolhido como um filho durante as primeiras fases deste trabalho.

Ao meu orientador professor Adelardo A. D. de Medeiros, pelos ensinamentos, confiança e compreensão dispensados durante todos os momentos e que me proporcionaram a conclusão desta dissertação.

Ao meu co-orientador Pablo Javier Alsina, por todos os ensinamentos, apoio e incentivo durante a minha graduação e para conclusão desta dissertação.

A todos os amigos e colegas de laboratório que trabalharam duro para permitir o desenvolvimento e concretização deste projeto.

Resumo

Esta dissertação trata do problema de controle dinâmico de robôs móveis não-holonômicos com duas rodas e acionamento diferencial. São apresentadas estratégias para controle de posição e de orientação de robôs com as características citadas, baseadas apenas em informações de configuração do robô (posição e orientação no espaço cartesiano x , y e θ), coletadas a partir de um sistema de posicionamento absoluto. O desenvolvimento da estratégia está associado a uma mudança nas variáveis controladas do robô, de x , y e θ para s (representando o deslocamento linear do robô) e θ , e na representação em coordenadas polares do modelo cinemático do robô. Desta forma, consegue-se uma representação linear para o modelo dinâmico do robô, a partir da qual são propostas as estratégias de controle. Com as estratégias definidas, mostra-se que é possível implementar controladores dinâmicos lineares para permitir a estabilização do sistema robótico. As estratégias apresentadas também fornecem flexibilidade com relação à escolha do tipo de controlador dinâmico (P, PI, PID, técnica *Model Matching*, entre outros) a ser utilizado. Esta dissertação apresenta uma introdução à robótica móvel e às características do tipo de sistema abordado, seguindo para uma revisão bibliográfica, desenvolvimento das estratégias de controle e projeto dos controladores. Finalmente, são apresentados e discutidos resultados obtidos através de simulações e resultados experimentais.

Abstract

This work addresses the dynamic control problem of two-wheeled differentially driven non-holonomic mobile robot. Strategies for robot positioning control and robot orientating control are presented. Such strategies just require information about the robot configuration (x , y and θ), which can be collected by an absolute positioning system. The strategies development is related to a change on the controlled variables for such systems, from x , y and θ to s (denoting the robot linear displacement) and θ , and makes use of the polar coordinates representation for the robot kinematic model. Thus, it is possible to obtain a linear representation for the mobile robot dynamic model and to develop such strategies. It is also presented that such strategies allow the use of linear controllers to solve the control problem. It is shown that there is flexibility to choice the linear controller (P, PI, PID, Model Matching techniques, others) to be implemented. This work presents an introduction to mobile robotics and their characteristics followed by the control strategies development and controllers design. Finally, simulated and experimental results are presented and commented.

Sumário

Lista de Figuras	vii
1 Introdução	1
1.1 Objetivo e Escopo do Trabalho	2
1.2 Robótica Móvel	2
1.3 Não-Holonomia de Robôs Móveis	4
1.4 Controle de Robôs Móveis	5
1.4.1 Métodos para Medição do Estado do Robô	6
1.4.2 Classificação do Problema de Controle	7
1.5 Organização da Dissertação	10
2 Revisão da Literatura	11
2.1 Controladores Cinemáticos	11
2.1.1 Controladores para Seguimento de Trajetória	12
2.1.2 Controladores Estabilizantes	15
2.2 Controladores Dinâmicos	18
2.3 Conclusões	20
3 Plataforma de Desenvolvimento	21
3.1 Descrição da Estrutura Física e Sistemas Computacionais	21
3.2 Modelagem do Robô	21
3.2.1 Modelagem Cinemática	22
3.2.2 Modelagem Dinâmica	27
3.2.3 Aspectos de Não-Holonomia	31
3.2.4 Aspectos de Controlabilidade	31
3.3 Simulador de Robôs Móveis	32

4	Estratégia de Controle	35
4.1	Introdução	35
4.2	Desenvolvimento da Estratégia de Controle	38
4.2.1	Transformação do Modelo Cinemático para Coordenadas Polares	38
4.2.2	Controlador de Posição	40
4.2.3	Controlador de Orientação	42
4.3	Resumo das Estratégias de Controle	50
4.4	Conclusão	51
5	Projeto e Implementação	52
5.1	Características do Sistema	52
5.2	Projeto dos Controladores	56
5.2.1	Controle de Posição	57
5.2.2	Controle de Orientação	61
5.3	Conclusão	63
6	Resultados	65
6.1	Controle de Posição	65
6.1.1	Controle de Posição com Formulação Original	66
6.1.2	Controle de Posição com Modificação	68
6.1.3	Controle de Posição para Seguimento de Trajetória	70
6.1.4	Controle de Posição com Ruídos e Incertezas	71
6.1.5	Resultados Experimentais para a Estratégia de Controle de Posição . . .	73
6.2	Controle de Orientação	75
6.2.1	Resultados Experimentais para Estratégia de Controle de Orientação . . .	87
6.3	Conclusão	90
7	Considerações Finais	91
	Referências Bibliográficas	93

Lista de Figuras

1.1	Níveis hierárquicos da arquitetura de navegação para robôs móveis.	3
1.2	Estacionamento de um robô móvel com restrições não-holonômicas.	5
1.3	Problema de estabilização.	8
1.4	Problema de seguimento de trajetória.	8
2.1	Simulação de seguimento de trajetória via DFL.	14
2.2	Simulação de seguimento de trajetória via NTI.	14
2.3	Simulação de estabilização via POL.	16
2.4	Simulação de estabilização via DFL.	16
3.1	Estrutura física da plataforma de desenvolvimento.	22
3.2	Representação esquemática do robô.	23
3.3	Composição de velocidades no robô.	24
3.4	Restrições não-holonômicas.	25
3.5	Análogo elétrico dos atuadores do robô móvel.	28
3.6	Ilustração do erro de quantização gerado pelo sistema de visão.	33
4.1	Diagrama de blocos do sistema de controle.	37
4.2	Definições envolvidas no problema de controle.	38
4.3	Diagrama de blocos da estratégia de controle de posição.	42
4.4	Diagrama de blocos da estratégia de controle de posição combinada a um gerador de trajetórias.	42
4.5	Idéia do controle de orientação baseado em referência móvel.	43
4.6	Definições envolvidas no problema de estabilização.	44
4.7	Exemplo do controle de orientação (a).	46
4.8	Exemplo do controle de orientação (b).	46
4.9	Exemplo do controle de orientação (c).	46

4.10	Exemplo do controle de orientação (d).	46
4.11	Exemplo do controle de orientação (e).	46
4.12	Exemplo de referência móvel em conjunto com controle de posição.	46
4.13	Exemplo onde os erros definidos não atendem ao problema de controle.	47
4.14	Exemplo onde o objetivo x_d e y_d encontram-se atrás do robô.	49
5.1	Diagrama de blocos do sistema de controle de posição utilizando controladores monovariáveis.	55
5.2	Resposta ao degrau da função de transferência G_s em malha fechada.	56
5.3	Resposta ao degrau da função de transferência G_θ em malha fechada.	56
5.4	(a) Resposta em x, y e θ do sistema robótico com realimentação de erros, sem controlador. (b) Resposta em s e θ do sistema robótico.	57
5.5	Lugar das raízes para $G_\theta(s)$	59
5.6	Lugar das raízes para $G_\theta(s)$ com controlador $C(s)$ tipo PID.	59
5.7	Resposta do sistema (movimento angular) em malha fechada, com e sem o controlador PID projetado.	59
5.8	Lugar das raízes para $G_s(s)$	60
5.9	Lugar das raízes para $G_s(s)$ com controlador $C(s)$ tipo PD.	60
5.10	Resposta do sistema (movimento linear) em malha fechada, com e sem o controlador PD projetado.	61
5.11	Evolução das variáveis de estado Δl , $\Delta \phi$ e ϕ para condição de oscilação.	62
5.12	Evolução das variáveis de estado Δl , $\Delta \phi$ e ϕ para condição sem oscilação.	62
5.13	Evolução das variáveis de estado x , y e θ para condição de oscilação.	62
5.14	Evolução das variáveis de estado x , y e θ para condição sem oscilação.	62
5.15	Diagrama de blocos do sistema de controle de orientação utilizando controladores monovariáveis.	63
6.1	Simulação do controle de posição - caso 1.	66
6.2	Simulação do controle de posição - caso 2.	67
6.3	Simulação de controle de posição com modificação proposta.	68
6.4	Trajetórias para uma mesma configuração desejada, com e sem a modificação na estratégia de controle.	69
6.5	Simulação demonstrando chaveamento do sinal de controle.	69
6.6	Simulação demonstrando sinal de controle suave.	70

6.7	Trajetória circular gerada e evolução do robô utilizando a estratégia de controle de posição.	70
6.8	Controle de posição com introdução de fontes de ruído e incertezas.	72
6.9	Sinal de controle na presença de ruídos e incertezas.	72
6.10	Sinal de controle na ausência de ruídos e incertezas.	72
6.11	Resultados experimentais para a estratégia de controle de posição.	73
6.12	Resultados experimentais para a estratégia de controle de posição aplicada ao seguimento de caminho.	74
6.13	Controle de orientação - caso 1.	76
6.14	Controle de orientação - caso 2.	77
6.15	Controle de orientação - caso 3.	78
6.16	Controle de orientação - caso 4.	79
6.17	Controle de orientação - caso 5.	80
6.18	Controle de orientação - caso 6.	81
6.19	Controle de orientação - caso 7.	82
6.20	Controle de orientação - caso 8.	83
6.21	Controle de orientação - caso 9.	84
6.22	Controle de orientação - caso 10.	85
6.23	Controle de orientação - caso 11.	86
6.24	Resultados experimentais controle de orientação - caso 1.	87
6.25	Resultados experimentais controle de orientação - caso 2.	88
6.26	Resultados experimentais controle de orientação - caso 3.	89

Capítulo 1

Introdução

Os sistemas robóticos podem ser classificados basicamente como robôs manipuladores e robôs móveis. Ambas as classes de sistemas robóticos vêm passando por avanços tecnológicos que elevam cada vez mais seus graus de autonomia e confiabilidade, refletindo em uma crescente aceitação pelo mercado (industrial, militar, residencial, dentre outros) destes sistemas.

A classe dos robôs móveis é constituída por robôs que têm a capacidade de locomoção no espaço tridimensional (robôs aeroespaciais, atmosféricos ou submarinos) ou em uma superfície (robôs terrestres). Os sistemas robóticos com capacidade de locomoção no espaço podem ser sistemas para exploração espacial, sistemas aéreos para controle ambiental ou espionagem, e sistemas submarinos para realização de tarefas subaquáticas. Os robôs com capacidade de locomoção em uma superfície são comumente utilizados para tarefas de exploração de terrenos, execução de atividades em ambientes hostis, ou tarefas de auxílio em ambientes industriais, hospitalares e residenciais. Desta forma, pode-se observar que há uma ampla área de aplicação para esta classe de robôs. Existem também robôs híbridos, isto é, compostos por manipuladores e plataformas móveis, como é o caso dos humanóides e veículos terrestres, aeroespaciais e submarinos dotados de braços manipuladores.

O foco deste trabalho está no desenvolvimento de uma estratégia de controle para sistemas robóticos móveis terrestres. O objetivo e escopo do trabalho são abordados em mais detalhes na seção a seguir. Em seguida são introduzidos os principais conceitos envolvidos no trabalho, a fim de permitir um melhor entendimento deste. A seção 1.2 apresenta a arquitetura típica encontrada em robôs móveis autônomos. Na seção 1.3 são apresentadas as características de um robô móvel não-holonômico. Uma abordagem ao problema de controle para robôs móveis é apresentada na seção 1.4. Por fim, a seção 1.5 apresenta a organização desta dissertação.

1.1 Objetivo e Escopo do Trabalho

O objetivo deste trabalho é o estudo e desenvolvimento de estratégias de controle de posição e orientação para robôs móveis com rodas. As definições mais precisas tanto do sistema abordado quanto do problema de controle serão dadas nas seções 1.3 e 1.4.

Neste nível, considerado o mais baixo na hierarquia do sistema robótico móvel, a presença de obstáculos não é considerada. Isto é, este nível é responsável apenas por garantir a execução de caminhos ou trajetórias gerados por níveis hierárquicos superiores. Na seção 1.2, é apresentado um detalhamento da hierarquia encontrada na arquitetura de robôs móveis autônomos.

A principal motivação para este trabalho reside no pequeno número de estratégias de controle dinâmico para resolução deste problema disponíveis na literatura. Além disso, a revisão bibliográfica mostra que a maioria das estratégias de controle para estes fins são baseadas em medições das velocidades angular e linear do robô, que nem sempre são de medição robusta e confiável, conforme será abordado no capítulo 2. Desta forma, busca-se, uma estratégia de controle que considere apenas uma parte do estado do robô (sua localização x , y e θ), mensurável através de um sistema de posicionamento absoluto. O sistema de posicionamento absoluto utilizado para desenvolvimento da estratégia de controle proposta neste trabalho é apresentado em detalhes no capítulo 3.

O estudo e desenvolvimento apresentados neste trabalho possuem aplicação direta em pesquisas e projetos em andamento no Laboratório de Sistemas Inteligentes da UFRN, que envolvem alunos de graduação e pós-graduação. Além da aplicação direta citada, a estratégia desenvolvida também poderá ser facilmente utilizada, devido à sua simplicidade de implementação, em sistemas robóticos similares aos considerados neste trabalho.

1.2 Robótica Móvel

Um robô móvel é composto por uma arquitetura de *hardware* e *software* que é responsável por garantir a execução correta de sua navegação para execução de uma determinada tarefa. Esta arquitetura para navegação de robôs móveis pode ser dividida em níveis hierárquicos, conforme apresentado na figura 1.1.

Cada um dos níveis apresentados na figura 1.1 constitui áreas de pesquisa que podem ser tratadas individualmente dentro do estudo da robótica. A seguir estes níveis serão brevemente descritos.

O nível *Percepção* representa a interface do robô com seu ambiente trabalho. Este nível é

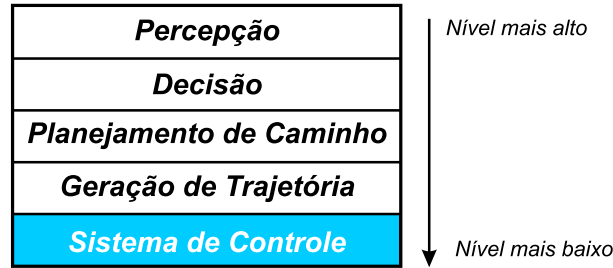


Figura 1.1: Níveis hierárquicos da arquitetura de navegação para robôs móveis.

composto por sensores e algoritmos para tratamento das informações coletadas. As informações são tratadas de forma a ter algum significado para o sistema robótico e para a tarefa que este necessita executar. Em seguida as informações são passadas para o nível de decisão. As informações extraídas do ambiente e do estado do robô são, normalmente, a localização do robô no ambiente, a localização de obstáculos e informações sobre a velocidade do robô.

O nível de *Decisão* é responsável pela inteligência e tomada de decisões, de acordo com as informações obtidas através do nível de percepção. Estas decisões variam com o tipo de tarefa que se deseja realizar. Normalmente são utilizadas, neste nível, árvores de decisão ou técnicas de inteligência artificial.

No nível de *Planejamento de Caminho* é feito o planejamento do melhor caminho que leva o robô de sua localização original a uma determinada localização desejada, levando em consideração os obstáculos e as dimensões do robô. Define-se como localização ou configuração do robô a soma das informações de posição e orientação. Há várias técnicas que podem ser utilizadas neste nível, como campos de potencial, diagramas de Voronoi, decomposição exata em células convexas, dentre outras.

O nível de *Geração de Trajetória* é responsável por determinar as restrições temporais para a execução de um determinado caminho, gerando, desta forma, uma trajetória. A geração de trajetórias impõe as velocidades e acelerações que o robô deve adotar em cada instante de tempo para que a execução do caminho seja realizada no período especificado. Uma das possíveis técnicas para geração de trajetórias é a utilização de polinômios de interpolação.

O nível *Sistemas de Controle* é responsável por garantir a execução das trajetórias através da geração de sinais de comando para os atuadores. Desta forma, este nível garante que as velocidades e acelerações impostas pelo nível de geração de trajetória sejam executadas de forma correta. Normalmente os sistemas de controle para robôs móveis são sistemas baseados em realimentação do estado do robô, isto é, informações a respeito da velocidade e localização do robô geradas pelo nível de percepção.

Os sinais de controle gerados pelo sistema de controle são comandos para os acionamentos dos atuadores do robô, que podem ser de origem eletrônica, pneumática, mecânica, dentre outros. Os atuadores também podem ser de diversos tipos, por exemplo, motores elétricos, propulsores a jato, ou motores a combustão. Os atuadores são conectados às partes móveis do robô que possibilitam a locomoção, como rodas, hélices, pás de uma turbina, discos de inércia, esteiras ou patas.

Pela descrição dos níveis hierárquicos da arquitetura de navegação, é possível notar o nível de complexidade que envolve o projeto e implementação de robôs móveis autônomos. Conforme destacado na figura 1.1 o nível de sistemas de controle é o objeto de estudo desta dissertação e será abordado em mais detalhes.

1.3 Não-Holonomia de Robôs Móveis

Os robôs móveis terrestres possuem a capacidade de realizar movimentações semelhantes às realizadas por veículos dotados de rodas, esteiras ou outros métodos de locomoção, que permitem o deslocamento em uma superfície. Desta forma, as características de movimentação de um robô dependem da disposição e funcionalidade de seus dispositivos de locomoção, que podem restringir seus movimentos ou a realização de trajetórias mais complexas. Em particular, alguns robôs móveis podem apresentar restrições de movimento denominadas restrições não-holonômicas.

O termo holonômico significa “universal”, “integral”, “integrável” (literalmente: holo = o todo, conjunto, totalidade - nomia = lei) e portanto, sistemas não-holonômicos podem ser interpretados como sistemas não-integráveis [6]. Definem-se como não-holonômicos sistemas com dimensão finita onde algum tipo de restrição é imposta a um ou mais estados do sistema. No caso de robôs móveis com acionamento diferencial a restrição é imposta pela impossibilidade do robô se movimentar em todas as direções, devido ao sistema não possuir atuadores que permitam tais movimentos, assim como pela condição de não-deslize (considera-se que não há derrapagem). Um robô tipo carro, ou seja, com um sistema de locomoção por rodas e com a mesma configuração de um carro, é um clássico exemplo de robô móvel não-holonômico. Uma restrição não-holonômica impede que o robô execute movimentos normais à superfície do corpo de suas rodas, quando não há deslizamento. A figura 1.2 mostra como as restrições não-holonômicas se apresentam em uma, aparentemente simples, tarefa de estacionamento.

Na figura 1.2, é possível notar que a trajetória mais intuitiva para o estacionamento, sem levar em consideração o sistema de locomoção, seria uma linha reta deslocando o robô lateralmente

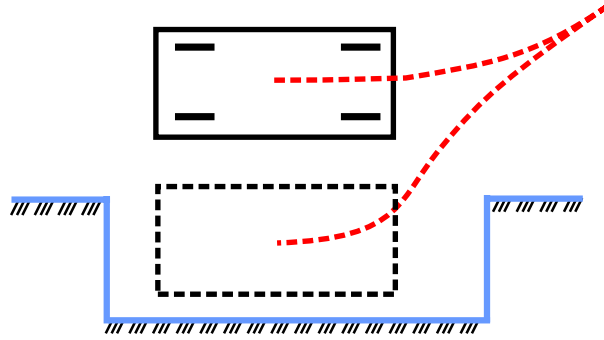


Figura 1.2: Estacionamento de um robô móvel com restrições não-holonômicas.

para a posição final desejada. No entanto, esta trajetória não é factível quando o sistema de locomoção (rodas em configuração tipo carro) é considerado. Desta forma, é necessário que seja gerada uma trajetória factível para a tarefa de estacionamento, levando em consideração as restrições não-holonômicas impostas pelo sistema de locomoção. Desta forma, é possível notar que a presença de restrições não holonômicas torna bem mais complexa a geração de trajetória para este tipo de tarefa, exigindo um maior processamento neste nível.

Deve-se notar que não apenas os robôs com configuração tipo carro apresentam restrições não-holonômicas. Um navio simples (isto é, que não é dotado de sistema de posicionamento dinâmico), por exemplo, é um sistema que apresenta este tipo de restrição de movimento. No entanto, não possui rodas e é um veículo marinho. Como descrito anteriormente, as condições que definem as restrições de movimento de um sistema robótico qualquer, são dadas pela configuração do seu sistema de locomoção. No capítulo 3 são apresentadas em mais detalhes as características dos robôs abordados, que são dotados de rodas e apresentam restrições não-holonômicas de movimento.

Na seção 1.4, a seguir, são descritos os problemas de controle de robôs móveis com rodas e restrições não-holonômicas, que constituem o tema principal deste trabalho. Também são introduzidos métodos para medição do estado do robô, que são necessários na implementação de controladores por realimentação, descritos a seguir.

1.4 Controle de Robôs Móveis

O problema de controle de robôs móveis não-holonômicos tem sido bastante estudado nos últimos anos por tratar-se de um problema complexo e com grande demanda para as diversas aplicações já citadas. Este tipo de sistema pertence a uma classe de sistemas não-lineares, multivariáveis, sub-atuados e com restrições não-holonômicas de movimento. Há na literatura

um vasto número de estratégias de controle que abordam esta classe de sistemas.

O principal objetivo de projetar-se um sistema de controle para robôs móveis é garantir a execução de uma determinada tarefa com um desempenho especificado. Os sistemas de controle por realimentação permitem garantir, com maior precisão, mesmo na presença de distúrbios, o bom desempenho do sistema controlado. Esta classe de controladores é caracterizada pela utilização do estado do sistema controlado, ou parte dele (apenas sua saída, por exemplo), o que permite ao controlador realizar ajustes no sinal de controle baseado no cálculo do erro entre a resposta temporal do sistema controlado e a resposta desejada. Desta forma, pode-se notar que é necessário, para fins de controle, obter medições precisas sobre o estado completo ou parcial do robô, dependendo da estratégia de controle utilizada (quanto maior for a quantidade de informações disponíveis, maior flexibilidade haverá para o projeto do sistema de controle).

1.4.1 Métodos para Medição do Estado do Robô

Nesta seção são introduzidos brevemente alguns métodos para reconstrução do estado do robô, ou parte dele, que é de fundamental importância na implementação de controladores por realimentação.

O estado do robô, denotado por \mathbf{x} , é constituído das seguintes variáveis de estado:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ \theta \\ v \\ w \end{bmatrix}$$

Onde x e y são a posição no espaço cartesiano, θ é a orientação do robô, v é a sua velocidade linear, e w sua velocidade angular. Os métodos existentes para reconstrução do estado \mathbf{x} podem ser divididos em dois grupos que se distinguem pelo tipo de sensores utilizados, a saber:

- **Sensores proprioceptivos:** são sensores que fornecem informações sobre o estado do robô baseado apenas em medições internas do robô. Em robôs móveis planares os sensores proprioceptivos mais utilizados são os *encoders* incrementais. Esses sensores fornecem a posição angular das rodas do robô, a partir da qual é derivada uma estimativa da odometria do robô e velocidades angulares. No entanto, as informações obtidas a partir deste tipo de sensor estão sujeitas a erros aleatórios e sistemáticos, devido ao deslizamento das rodas durante manobras, diferença no tamanho das rodas e erros de integração numérica. Outros

exemplos de sensores proprioceptivos são os acelerômetros e giroscópios, mais utilizados em sistemas robóticos autônomos que movimentam-se em três dimensões.

- **Sensores externoceptivos:** são sensores que fornecem informações sobre o estado do robô baseado em dados coletados do ambiente. Os sensores externoceptivos mais utilizados em robótica são câmeras, sonares, laser e GPS. Este tipo de sensor fornece um dado muito mais preciso (normalmente a posição absoluta do robô), pois se baseia em pontos fixos no ambiente a partir dos quais realiza cálculos sobre o deslocamento relativo do robô (através de triangulação, por exemplo). Apesar de apresentar melhores resultados, esta classe de sensores normalmente requer um pesado processamento de imagens, tornando a sua utilização, em determinados casos (por exemplo, baixo poder de processamento das CPUs embarcadas), inviável. Os sensores externoceptivos não precisam estar necessariamente embarcados na plataforma do robô móvel. Desta forma, também é possível utilizar um sistema de visão, externo ao robô, que seja responsável por observá-lo em seu espaço de trabalho e informar a uma unidade de processamento *on-board*, ou *off-board* a sua posição absoluta. As limitações deste método repousam no fato de que o espaço de trabalho fica restrito ao campo de visão da câmera. É importante lembrar que estes sensores fornecem a localização do robô com uma boa precisão, no entanto, a tentativa de derivar a posição para obtenção da velocidade linear e angular do robô não é um método robusto.

Deve-se notar que é perfeitamente possível a fusão da informação proveniente de dois ou mais sensores, mesmo pertencendo a classes diferentes. Normalmente combinações são feitas entre sensores proprioceptivos e externoceptivos a fim de obter correções periódicas dos erros acumulados por sensores da primeira classe.

1.4.2 Classificação do Problema de Controle

Para um melhor entendimento das estratégias de controle analisadas no capítulo 2, uma breve introdução aos típicos problemas de controle de robôs móveis é apresentada a seguir. Estes problemas podem ser divididos em duas classes principais:

- **Problema de Estabilização:** O problema de estabilização consiste em atingir uma configuração final desejada para o robô, que é definida por uma localização no espaço cartesiano e uma determinada orientação, a partir de uma configuração inicial qualquer do robô.
- **Problema de Seguimento de Trajetória:** O problema de seguimento de trajetória consiste em fazer com que o robô atinja e siga uma determinada trajetória no espaço

cartesiano (i.e, um caminho geométrico com uma lei temporal associada), iniciando de uma determinada configuração inicial, dentro ou fora da trajetória.

Um outro tipo de problema é o seguimento de caminho, que pode ser considerado como um problema de seguimento de trajetória porém com a ausência da lei temporal. Desta forma, o robô deve apenas seguir um determinado caminho geométrico sem restrições temporais. Como normalmente os controladores são divididos entre estabilizantes e rastreadores (para seguimento de trajetória), não foi definida nesta seção a classe de problemas de seguimento de caminho. As figuras 1.3 e 1.4 apresentam uma representação gráfica destas classes.

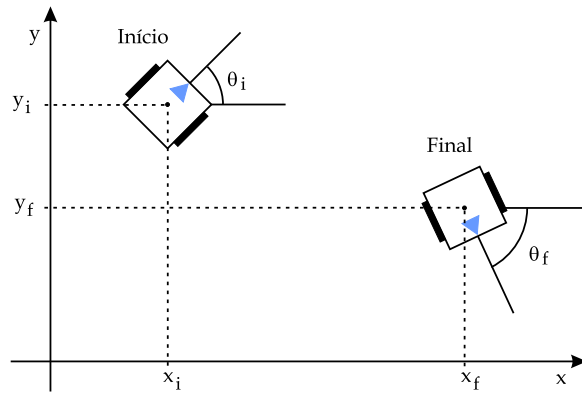


Figura 1.3: Problema de estabilização.

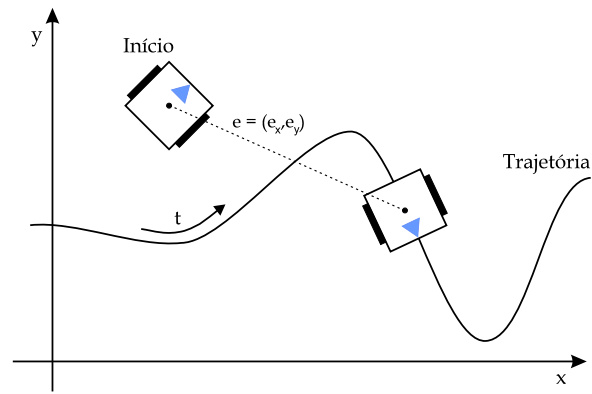


Figura 1.4: Problema de seguimento de trajetória.

Na tarefa de seguimento de trajetória, o robô deve seguir um determinado caminho cartesiano com uma específica lei temporal associada, equivalentemente, o robô deve rastrear um “robô referência” que se move com o tempo. Desta forma, o problema torna-se mais simples, pois passa a ser um problema de estabilização do erro e (figura 1.4) entre a configuração cartesiana do robô e da referência. Como será apresentado mais adiante, o problema de estabilização de robôs móveis é mais complexo do que o problema de seguimento de trajetória. No primeiro caso, é necessário atingir três objetivos de controle (x_d , y_d e θ_d), enquanto no segundo caso há apenas dois objetivos ($x_d(t)$ e $y_d(t)$), isto é, supondo que a trajetória gerada leva em conta as restrições não-holonômicas do robô (trajetória factível).

Há na literatura controladores que abordam apenas o problema de seguimento de trajetória, por ser mais simples. No capítulo 2 os tipos de controladores para os problemas de estabilização e seguimento de trajetória serão comentados em mais detalhes.

Em termos de teoria de controle, pode-se definir o problema de estabilização como um problema de regulação, enquanto que um problema de seguimento de trajetória pode ser definido como um problema de servo-controle. No entanto, o problema de estabilização torna-se mais complexo devido à inexistência, *à priori*, de uma trajetória que respeite as restrições não-holonômicas do

robô.

O projeto de sistemas de controle para o problema de estabilização ou seguimento de trajetória pode ser abordado de duas formas, uma considerando apenas o modelo cinemático do robô, e outra levando em consideração os modelos cinemático e dinâmico do robô. O modelo cinemático do robô representa as características de movimento e as restrições destes. O modelo cinemático também é um modelo dinâmico, pois o estado do robô, definido neste modelo, também varia com as excitações do sistema e depende do estado no instante imediatamente anterior, porém este não inclui as forças dinâmicas que atuam sobre o robô, daí a separação entre modelos cinemático e dinâmico. O modelo dinâmico representa como o robô responde às entradas de controle externas no decorrer do tempo, levando em conta as forças dinâmicas atuando sobre sua estrutura, seu momento de inércia, o modelo dinâmico dos atuadores do sistema e as forças de atrito envolvidas.

As estratégias de controle baseadas apenas em modelos cinemáticos são derivadas considerando que a parte mecânica do sistema robótico responde instantaneamente (sem transitório) ao sinal de controle u aplicado (usualmente um sinal elétrico). Esta simplificação é comumente assumida quando há algum tipo de sistema de controle projetado localmente para os atuadores do sistema. No entanto, essas estratégias tornam-se frágeis quando não há um sistema de controle específico para os atuadores. Quando isto ocorre não há garantia do funcionamento correto do controlador cinemático, haja vista que o robô não responde instantaneamente aos sinais de controle gerados (torques ou velocidades) e aplicados aos seus atuadores, devido à dinâmica do robô e dos próprios atuadores (momento de inércia, massa, coeficientes de atrito, etc). Desta forma, é interessante projetar controladores que levem em conta o modelo completo do robô (modelos cinemático e dinâmico), tornando-os mais abrangentes que os controladores puramente cinemáticos. Conforme será apresentado no capítulo 2, há controladores dinâmicos adaptativos que, além de realizar a compensação dinâmica, são capazes de lidar com variações paramétricas que possam ocorrer durante a operação do robô, e não necessitam das informações de parâmetros dinâmicos do robô para sua implementação. Um exemplo clássico de variação paramétrica é a adição de carga ao robô, que leva ao aumento de sua massa e pode alterar o seu momento de inércia e todas as outras características dinâmicas envolvidas.

Portanto, os controladores que consideram apenas o modelo cinemático, são denominados controladores cinemáticos, enquanto que os que consideram os modelos cinemático e dinâmico são denominados controladores dinâmicos. No capítulo 2 serão apresentados e analisados controladores de ambas as classes encontrados na literatura, assim como alguns resultados simulados

obtidos de suas implementações.

1.5 Organização da Dissertação

No capítulo 2 é apresentado um estudo bibliográfico das principais estratégias de controle cinemático e dinâmico encontradas na literatura, contendo alguns resultados obtidos via simulação, e discussões sobre as estratégias apresentadas.

No capítulo 3 é apresentada a plataforma de desenvolvimento utilizada neste trabalho. São apresentados os sistemas que compõem a plataforma, a modelagem completa do robô em estudo, discussões sobre controlabilidade e restrições de movimento do robô, e a plataforma de simulação desenvolvida para realização de testes.

O capítulo 4 aborda a estratégia de controle desenvolvida para robôs móveis com rodas e restrições não-holonômicas de movimento. São apresentadas a análise do modelo do robô, a derivação da estratégia de controle de posição e a derivação para obtenção de uma estratégia de estabilização dinâmica.

No capítulo 5 é apresentado o projeto e implementação do controlador para o robô, baseado na estratégia desenvolvida. Também são apresentados os resultados do desempenho esperado, obtido via simulação.

O capítulo 6 inclui os resultados experimentais obtidos a partir da aplicação da estratégia de controle desenvolvida. Os resultados são discutidos, avaliados e comparados com os resultados obtidos via simulação.

Finalmente, no capítulo 7 são apresentadas as principais contribuições e conclusões do trabalho desenvolvido e perspectivas para trabalhos futuros.

Capítulo 2

Revisão da Literatura

Este capítulo aborda primeiramente alguns dos controladores cinemáticos mais citados na literatura. Estes são dispostos em dois grupos - controladores para seguimento de trajetória e controladores estabilizantes. Em seguida são abordadas algumas estratégias de controle dinâmico encontradas na literatura.

2.1 Controladores Cinemáticos

As estratégias de controle cinemático, conforme apresentação anterior (1.4.2), são baseadas na simplificação de que o robô responde instantaneamente ao sinal de controle gerado pelo controlador, isto é, não são considerados os efeitos dinâmicos resultantes do movimento do sistema robótico. Com esta simplificação o modelo que descreve o comportamento do robô é dado por:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2.1)$$

onde x e y representam a posição do robô no espaço cartesiano, θ representa o ângulo do robô em relação ao eixo das ordenadas, e v e ω representam as velocidades linear e angular do robô, respectivamente. Define-se uma configuração do robô pelo vetor q , tal que:

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (2.2)$$

As estratégias de controle para o problema de seguimento de trajetória e para o problema de estabilização são apresentadas separadamente nas seções que seguem.

2.1.1 Controladores para Seguimento de Trajetória

Na literatura são encontradas diversas estratégias de controle que abordam o problema de seguimento de trajetória [17, 5, 14, 23]. O controle de trajetória pode ser encarado como um problema mais simples do que o problema de estabilização, considerando que a trajetória gerada respeita as restrições não-holonômicas do robô. Isto se deve, principalmente, ao fato de que, corrigindo o erro de posição em relação ao tempo através do seguimento de uma referência móvel, o robô terá sua orientação automaticamente corrigida. Desta forma, para implementar um controlador de seguimento de trajetória, é necessário apenas projetar um bom controlador de posição e passar referências $x_d(t)$ e $y_d(t)$ variantes no tempo, de acordo com a trajetória a ser seguida. Tem-se que a orientação associada do robô é dada por $\theta_d = \tan^{-1}(\frac{\dot{y}_d(t)}{\dot{x}_d(t)})$. Uma estratégia para estabilização também poderia ser utilizada, caso a trajetória gerada inclua informações sobre orientação desejada para o robô ao longo do tempo. No entanto, também seria viável a utilização de um controlador de posição (i.e., não controla a orientação do robô, apenas a posição relativa x e y), de construção ainda mais simples. As referências $(x_d(t), y_d(t))$ para o controlador de trajetória podem ser traduzidas em comandos de velocidade linear e angular da seguinte forma (Luca et. al [12]):

$$v_d(t) = \pm \sqrt{\dot{x}_d^2(t) + \dot{y}_d^2(t)}$$

$$w_d(t) = \frac{\ddot{y}_d(t)\dot{x}_d(t) - \ddot{x}_d(t)\dot{y}_d(t)}{\dot{x}_d^2(t) + \dot{y}_d^2(t)}$$

A seguir são abordados alguns controladores encontrados na literatura para a tarefa de seguimento de trajetória.

- **Linearização por Realimentação Dinâmica** (DFL – *Dynamic Feedback Linearization*) [5]: é uma estratégia de controle bastante conhecida para o problema de seguimento de trajetórias. Deve-se notar que a realimentação dinâmica, neste contexto, não se refere ao modelo dinâmico do robô. O objetivo desta estratégia é atingir a linearização exata para o modelo do robô, através da realimentação de dinâmicas acrescentadas (aumentando o número de variáveis de estado) ao sistema. A partir da linearização obtida é possível projetar leis de controle lineares que atendam ao objetivo desejado. Para o modelo cinemático a linearização exata pode ser obtida adicionando apenas um estado ao robô [12], isto é,

adicionando-se um integrador na entrada de velocidade linear

$$v = \xi \quad \dot{\xi} = a$$

e fazendo uma redefinição das entradas do sistema (a, w) como

$$\begin{bmatrix} a \\ w \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\frac{\sin \theta}{\xi} & \frac{\cos \theta}{\xi} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (2.3)$$

A estabilização pode então ser obtida através da lei de controle

$$\begin{cases} u_1 = \ddot{x}_d + K_{p1}(x_d - x) + K_{d1}(\dot{x}_d - \dot{x}) \\ u_2 = \ddot{y}_d + K_{p2}(y_d - y) + K_{d2}(\dot{y}_d - \dot{y}) \end{cases} \quad (2.4)$$

com $K_{pi} > 0$, $K_{di} > 0$ ($i = 1, 2$). O estado inicial do integrador deve ser $\xi \neq 0$ e a trajetória desejada deve ser persistente a fim de evitar singularidades da equação 2.3. Uma condição necessária para o exato seguimento das trajetórias é que estas sejam duplamente deriváveis. Na perda de suavidade da trajetória, o robô desviará da trajetória desejada. No entanto, se essa descontinuidade for momentânea, o controlador é capaz de levar o robô ao restante da trajetória suave a uma taxa exponencial. Deve-se observar que, para implementar este controlador, é necessário possuir sensores capazes de informar a posição do robô e velocidade linear de deslocamento no plano cartesiano. A figura 2.1 apresenta uma simulação de seguimento de trajetória utilizando a estratégia de controle DFL. Para esta simulação a configuração inicial do robô foi $q = \begin{bmatrix} 0.8 & -0.2 & 0 \end{bmatrix}^T$, os ganhos utilizados para o controlador foram $K_{pi} = 0.5$, $K_{di} = 1$ ($i = 1, 2$) e a trajetória foi gerada pela equação

$$\begin{cases} x_d(t) = k_3 \left[k_1 \cos(k_2 t) - \left(\frac{k_1}{4} \right) \cos(9k_2 t) \right] \\ y_d(t) = k_3 \left[k_1 \sin(k_2 t) - \left(\frac{k_1}{4} \right) \sin(9k_2 t) \right] \end{cases} \quad (2.5)$$

com $k_1 = 0.04$, $k_2 = 0.3$ e $k_3 = 15$.

- **Controle não-linear invariante no tempo** (NTI – *Nonlinear time-invariant control*) [17]: este controlador é resultado de uma abordagem de projeto de controladores para sistemas não-lineares via Lyapunov. Neste tipo de projeto busca-se, a partir da definição de uma função candidata de Lyapunov, uma lei de controle capaz de provar que a função candidata é, de fato, uma função de Lyapunov. A seguir é apresentada a lei de controle obtida [17]:

$$\begin{cases} v = v_d \cos(\theta_d - \theta) + k_1[(x_d - x) \cos \theta + (y_d - y) \sin \theta] \\ w = w_d + k_2 v_d \frac{\sin(\theta_d - \theta)}{\theta_d - \theta} [(y_d - y) \cos \theta - (x_d - x) \sin \theta] + k_3(\theta_d - \theta) \end{cases} \quad (2.6)$$

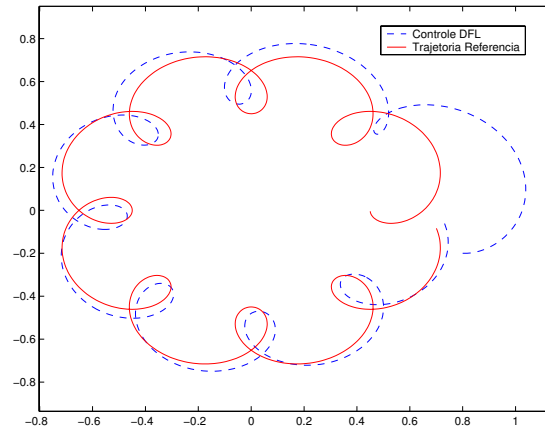


Figura 2.1: Simulação de seguimento de trajetória via DFL.

Uma escolha usual para os ganhos de controle é

$$k_1 = k_3 = 2\zeta\sqrt{w_d^2(t) + bv_d^2(t)}, \quad k_2 = b,$$

onde o coeficiente de amortecimento $\zeta \in (0,1)$ e $b > 0$. Pode-se perceber que esta estratégia apresenta a desvantagem de necessitar da informação θ_d , porém, não necessita de informações sobre as velocidades lineares de deslocamento no plano cartesiano. Na figura 2.2 é apresentada uma simulação realizada utilizando a estratégia NTI. Para esta simulação a configuração inicial adotada para o robô foi $q = \begin{bmatrix} 0.8 & -0.2 & 0 \end{bmatrix}^T$. Os parâmetros adotados na lei de controle foram $b = 50$ e $\zeta = 1$. A trajetória utilizada foi a mesma da equação 2.5.

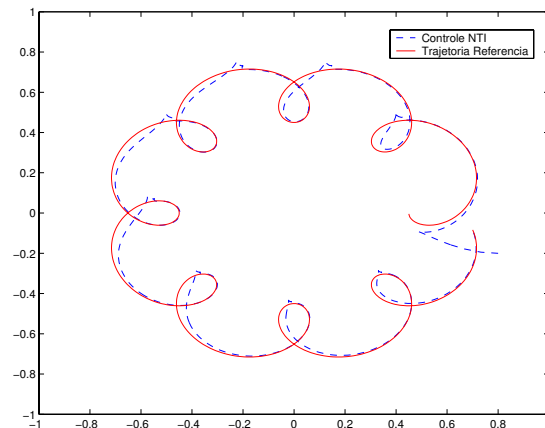


Figura 2.2: Simulação de seguimento de trajetória via NTI.

As estratégias para seguimento de trajetória descritas podem apresentar melhores resultados através de um melhor ajuste dos parâmetros envolvidos na lei de controle. Os parâmetros utilizados nas simulações foram puramente empíricos. É importante lembrar que, normalmente,

em uma arquitetura para controle de robôs móveis, a geração da trajetória é responsabilidade de um nível superior ao nível de controle. No nível de geração de trajetória, devem ser levadas em consideração todas as características do robô (restrições não-holonômicas, saturação dos sinais de controle e dinâmica do robô) de forma que as trajetórias geradas por este nível sejam factíveis.

2.1.2 Controladores Estabilizantes

O problema de estabilização, como já comentado, é mais complexo que o problema de seguimento de trajetória. Neste caso, é necessário recorrer a leis de controle variantes no tempo ou leis não-lineares descontínuas com realimentação para atingir os objetivos de controle, conforme será descrito no capítulo 3, seção 3.2.4. A seguir são introduzidos controladores estabilizantes de simples implementação, lembrando que estes abordam apenas o modelo cinemático do robô. Nos exemplos que seguem, a configuração desejada adotada é a origem do espaço de estado do robô ($x_d = 0, y_d = 0$ e $\theta_d = 0$).

- **Controle por Coordenadas Polares** (POL – *Polar coordinate control*): este controlador, assim como o NTI, também foi derivado a partir de técnicas de projeto de controladores não-lineares via Lyapunov. A estratégia POL [1] faz uso de uma transformação para coordenadas polares da configuração do robô. Esta transformação é singular na origem, como será mostrado a seguir.

$$\begin{cases} \rho = \sqrt{x^2 + y^2} \\ \gamma = \tan^{-1}\{y/x\} \\ \delta = \gamma + \theta \end{cases} \quad (2.7)$$

A lei de controle é dada por

$$\begin{cases} v = k_1 \rho \cos \gamma \\ w = k_2 \gamma + k_1 \frac{\sin \gamma \cos \gamma}{\gamma} (\gamma + k_3 \delta) \end{cases} \quad (2.8)$$

onde k_1 , k_2 e k_3 são constantes positivas. A transformação é singular na origem devido a função $\tan^{-1}(\cdot)$ não ser definida na origem. É interessante notar que esta estratégia exige apenas o conhecimento da configuração do robô no espaço de estado. A figura 2.3 apresenta uma simulação para o caso de estabilização utilizando a estratégia de controle POL. Os parâmetros usados na simulação foram $k_1 = 1$, $k_2 = 2$ e $k_3 = 3$. Considera-se o caso de levar o robô da configuração inicial $q = \begin{bmatrix} -0.2 & 0.5 & 0 \end{bmatrix}^T$ para a configuração final desejada $q_d = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$.

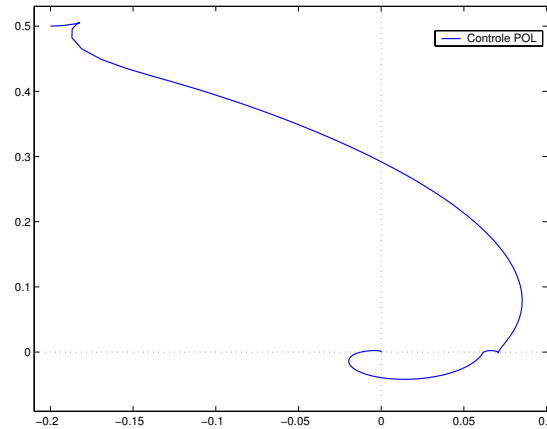


Figura 2.3: Simulação de estabilização via POL.

- **Linearização por Realimentação Dinâmica:** esta estratégia é a mesma utilizada para o seguimento de trajetória. Desta forma, para que o objetivo de controle seja modificado, basta fazer $\dot{x}_d = \dot{y}_d = \ddot{x}_d = \ddot{y}_d = 0$. A figura 2.4 apresenta uma simulação para o problema de estabilização via estratégia DFL. Os parâmetros utilizados na simulação foram $K_{pi} = 0.5$, $K_{di} = 1$ ($i = 1, 2$). Os estados inicial e final foram os mesmos especificados na simulação para a estratégia POL. É possível notar que este controlador não atinge todos os objetivos do problema de estabilização, pois apesar de atingir a origem, não necessariamente a atinge com a orientação desejada. Além disso este controlador necessita de informações não apenas de localização, mas também das velocidades lineares \dot{x} e \dot{y} , cujas medições normalmente são ruidosas.

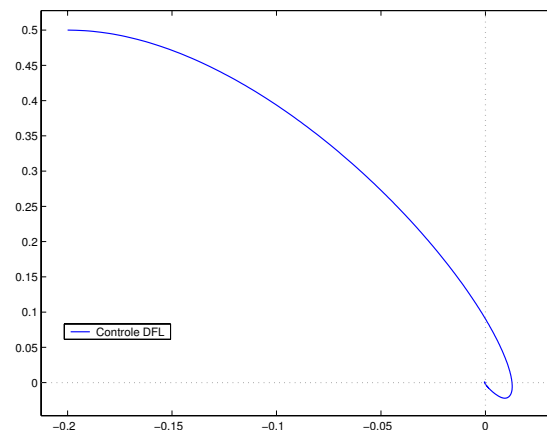


Figura 2.4: Simulação de estabilização via DFL.

Pode-se observar que o controlador POL é mais adequado para a tarefa de estabilização, pois atinge todos os objetivos de controle do problema de estabilização. Deve-se lembrar que o desempenho dos controladores pode ser otimizado através da sintonia de seus parâmetros de

controle. Os valores utilizados nas simulações foram escolhidos de forma empírica.

Uma abordagem experimental comparativa a respeito das estratégias apresentadas é encontrada no artigo de Luca et al., 2002 [11]. Neste artigo são avaliados os controladores apresentados com relação aos tipos de sensores utilizados para reconstrução do estado do robô.

Derivações de estratégias de controle cinemático por realimentação para robôs tipo carro podem ser encontradas em Luca et al., 1998 [12]. São detalhados aspectos de controlabilidade, transformação de modelos para a forma encadeada, e abordadas as estratégias mais conhecidas para estabilização e seguimento de trajetória de sistemas tipo carro.

Em Lizarralde, 1998 [10], é desenvolvida uma estratégia para controle cinemático de sistemas robóticos não-holonômicos, baseada em um método de otimização do tipo Newton. A estratégia apresentada consiste em um método, semelhante às técnicas utilizadas para controle preditivo (MPC), capaz de prever quais sinais de controle minimizam o erro para o seguimento de uma trajetória ou para atingir uma determinada configuração. Uma derivação desta lei para compensação dinâmica também é apresentada. No entanto, o algoritmo para predição da trajetória que estabiliza o robô em uma configuração desejada deve ser executado *off-line*.

Em Gaudiano et al., 1995 [7], apresenta-se uma abordagem ao problema de seguimento de trajetórias através de uma arquitetura composta por uma rede neural não-supervisionada, capaz de aprender sobre a odometria direta e reversa do robô. Após a fase de aprendizado, o sistema de controle neural é capaz de levar o robô a um estado arbitrário (posição apenas) ou a seguir uma determinada trajetória. O sistema mostra-se robusto a ruídos e incertezas do modelo cinemático. A utilização de redes neurais para controle de robôs móveis pode ser questionada devido ao tempo de treinamento necessário para a sua aplicação de forma eficiente. Apesar de existirem técnicas de aprendizado *on-line* para aplicação em sistemas de tempo real, o sistema de controle necessita de um determinado período para estabilização dos pesos sinápticos da rede neural, permitindo, desta forma, a ocorrência de falhas ou desempenho insatisfatório durante este período.

Em Yang e Kim, 1998 [23], é desenvolvida uma estratégia via modos deslizantes para controle de robôs móveis não-holonômicos. A estratégia apresentada propõe apenas a convergência do robô para uma trajetória de referência com estabilidade assintótica. Esta abordagem consiste de uma linearização por realimentação da equação dinâmica do robô e um controle por modos deslizantes para realizar o seguimento robusto da trajetória do erro dinâmico do sistema. A estratégia é frágil do ponto de vista que necessita de informações a respeito das velocidades linear e angular do robô, assim como de suas derivadas.

2.2 Controladores Dinâmicos

Muitas das estratégias de controle para compensação da dinâmica de robôs móveis não-holonômicos são baseadas em duas malhas de controle, uma interna, responsável pelo controle dinâmico, e outra externa, responsável pelo controle cinemático. Desta forma, é comum encontrar sistemas de controle que utilizam estratégias cinemáticas (como DFL e POL) para complementar a compensação dinâmica. A seguir são comentados alguns artigos encontrados na literatura sobre o tema.

Em Lages e Hemerly, 2000 [9], é proposto um esquema de controle em duas malhas. Uma malha interna, onde os parâmetros dinâmicos do robô são estimados e uma estratégia de controle linearizante adaptativo por modelo de referência é utilizada para fazer a compensação dinâmica, e outra malha externa, que utiliza um controlador cinemático baseado na transformação em coordenadas polares. Esta última assemelha-se bastante ao controlador POL [1] descrito anteriormente. Neste artigo são apresentados apenas resultados simulados para o problema de estabilização. Esta abordagem é interessante do ponto de vista de que não é necessário o conhecimento do modelo dinâmico do robô para sua implementação. Por outro lado, são necessárias as informações sobre as velocidades linear e angular do robô.

Em Oliveira e Lages, 2001 [14], é proposto um controlador com duas malhas de realimentação. A malha interna, responsável pela compensação dinâmica, é constituída por uma rede neural multi-camadas, com aprendizado *on-line*. A malha externa, responsável pelo controle cinemático, é baseada em uma transformação descontínua e é semelhante à estratégia proposta pelo controlador POL [1]. O treinamento da rede neural requer informações sobre as velocidades linear e angular do robô e suas derivadas. Resultados experimentais são apresentados apenas para seguimento de trajetória. O controlador apresentado não é capaz de resolver o problema de estabilização. A desvantagem de abordagens que fazem uso de redes neurais para fins de controle é que estas são de implementação mais complexa, dificultando o teste em outras plataformas robóticas.

Em Pereira, 2000 [15], é proposto um controlador baseado em um modelo dinâmico aproximado, obtido através de técnicas de modelagem e identificação de sistemas não-lineares. A partir do modelo identificado são projetados controladores PID que atendem aos requisitos de desempenho. A malha de controle também apresenta um bloco preditor a fim de atenuar o atraso puro de tempo, visto que, na plataforma considerada, os sinais de controle são gerados por um computador externo e transmitidos, via rádio, para o robô. Dependendo da implementação, este tipo de estratégia pode tornar-se sensível a determinadas tarefas pois baseia-se na utilização

de técnicas monovariáveis para controlar um sistema multivariável.

Em Tanner e Kyriakopoulos, 2002 [18], é apresentada uma lei de controle baseada em uma estratégia de *backstepping* descontínuo. O artigo trata primeiramente da derivação de uma lei de controle descontínua que atende à compensação cinemática para o problema de estabilização. Considerando a equação dinâmica do robô, os autores utilizam uma realimentação, levando a uma linearização da dinâmica do robô. Caso a lei de controle cinemática projetada fosse suave, seria possível a partir desta, achar uma lei de controle dinâmico realizando o *backstepping* através dos integradores do sistema. Isto é, a integração da lei de controle cinemático, juntamente com a realimentação para linearização, seria capaz de estabilizar a dinâmica do sistema completamente. Desta forma, os autores derivam uma expansão da técnica de *backstepping* via integradores para lidar com o caso no qual a lei de controle original que estabiliza o subsistema não é suave (neste caso, descontínua). O artigo apresenta simulações para o caso de estabilização. Este esquema de controle requer informações das velocidades linear e angular do sistema. Apesar de apresentar um bom desempenho (trajetórias suaves para estabilização e sinais de controle com baixa amplitude e chaveamento suave), a obtenção da lei de controle final não é trivial.

Dada a necessidade de atingir um objetivo de controle multivariável, definido pelo problema de estabilização, foi realizado um estudo sobre controle preditivo generalizado (GPC - *Generalized Predictive Control*) e sua aplicação à robótica móvel. O principal atrativo apresentado por controladores GPC é o fato de serem baseados na minimização de uma função de energia que leva em consideração a resposta do sistema (no caso a posição e orientação do robô) e os sinais de controle envolvidos. Desta forma, torna-se uma estratégia multivariável bastante interessante e muito difundida na indústria. Além disso, o GPC é um controlador baseado em modelo, isto é, é necessário o conhecimento do modelo da planta, principalmente devido a necessidade de realizar predições. Sendo assim, é possível realizar implementações adaptativas através da utilização de um identificador de sistemas *on-line*.

No entanto, verificou-se que para atingir uma implementação do GPC que levasse a bons resultados, seria necessária a adoção de uma identificação não-linear para o sistema. Verificou-se também [4] a necessidade de imposição de restrições (relaxamentos nos objetivos de controle) na aplicação do GPC para sistemas sub-atuados. O sistema robótico adotado para o desenvolvimento deste trabalho é um sistema sub-atuado, pois possui três variáveis de controle e apenas dois sinais de entrada para o sistema. Desta forma, não foi feita uma investigação mais profunda da utilização desta estratégia de controle para o sistema abordado.

Em trabalhos futuros deve ser investigada em mais detalhes a utilização de um controla-

dor adaptativo preditivo baseado na estratégia GPC. Visto que esta técnica permite unir a identificação *online* do modelo do robô com o sistema de controle.

2.3 Conclusões

A partir do estudo realizado pode-se notar que os controladores cinemáticos apresentados são comumente utilizados em conjunto com controladores dinâmicos em malhas distintas. Pode-se observar que a maioria das técnicas desenvolvidas para o problema de estabilização dinâmica são de complexa implementação e fazem uso das informações das velocidades linear e angular do robô, ou até mesmo suas derivadas.

Na maioria das metodologias apresentadas não há métodos formais para ajuste do desempenho do controlador, ou como o ajuste dos ganhos afeta o desempenho do sistema. Nos artigos apresentados a maioria dos ajustes são realizadas de forma empírica.

Com base na avaliação dos pontos supracitados encontra-se a motivação para o desenvolvimento de uma estratégia completa de controle (envolvendo a cinemática e dinâmica do robô), baseada apenas em informações de configuração, e com algoritmo de controle de implementação simples (capaz de ser facilmente difundido).

Capítulo 3

Plataforma de Desenvolvimento

Neste capítulo é abordada a plataforma sob a qual foi realizado o desenvolvimento deste trabalho, que inclui uma estrutura física, sistemas de computação e o sistema robótico. Na seção 3.1 são descritos a estrutura e os sistemas básicos que compõem a plataforma. Na seção 3.2 são derivados os modelos cinemático e dinâmico do robô móvel abordado, e são discutidos os aspectos de não-holonomia e controlabilidade. Na seção 3.3 é feita uma breve descrição do simulador desenvolvido para verificação de desempenho de estratégias de controle para robôs móveis não-holonômicos. Apesar da disponibilidade de uma plataforma experimental e de um simulador desenvolvido para o sistema [22], um outro simulador, capaz de reproduzir os principais efeitos e sistemas envolvidos no ambiente real, também foi utilizado para verificação de resultados.

3.1 Descrição da Estrutura Física e Sistemas Computacionais

A estrutura física utilizada é similar à apresentada na figura 3.1. Esta estrutura é constituída de uma área base (ou espaço de trabalho do robô), de um suporte para utilização de câmeras digitais para fins de localização do robô, de computadores pessoais para processamento de imagens e execução dos sistemas computacionais que controlam o robô, e uma estrutura de comunicação via rádio que permite o envio de sinais de controle do computador para o robô (mais detalhes em Vieira et al., 2001 [19]).

3.2 Modelagem do Robô

O modelo de um robô pode ser dividido em duas partes, um modelo cinemático e outro dinâmico. O modelo cinemático do robô representa as características de movimento e as restrições destes. Como já descrito nesta dissertação, o modelo cinemático também é um modelo

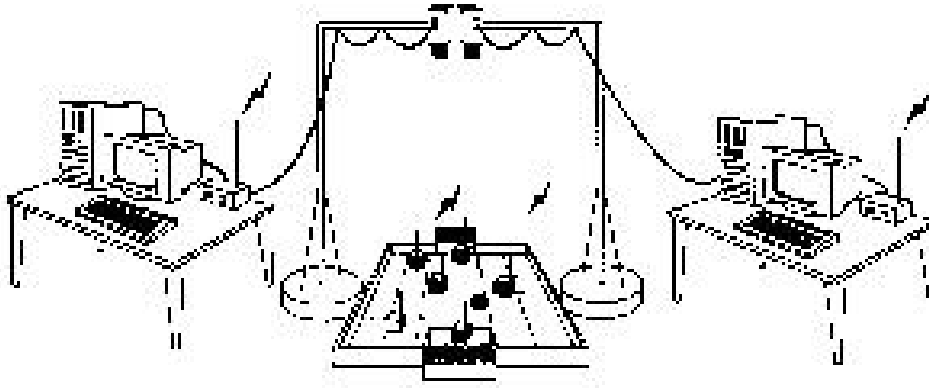


Figura 3.1: Estrutura física da plataforma de desenvolvimento.

dinâmico, pois o estado do robô, definido por este modelo, varia com as excitações de entrada e depende do estado no instante anterior, porém este não inclui as forças dinâmicas que atuam sobre o robô, daí a separação entre modelos cinemático e dinâmico. O modelo dinâmico representa como o robô responde às entradas de controle externas no decorrer do tempo, levando em conta as forças dinâmicas atuando sobre sua estrutura, seu momento de inércia, o modelo dinâmico dos atuadores do sistema e as forças de atrito envolvidas.

A maioria dos trabalhos relacionados à modelagem e ao controle de robôs móveis, encontrados na literatura, apresentam apenas uma abordagem ao modelo cinemático do robô, desprezando assim, a dinâmica dos atuadores e do robô. Há poucos artigos na literatura que abordam o modelo dinâmico para controle de robôs móveis [18, 9, 14, 16, 23, 10]. A seguir serão introduzidos os modelos cinemático e dinâmico de um robô móvel, a partir dos quais serão discutidos aspectos de restrições de movimento e de controlabilidade.

3.2.1 Modelagem Cinemática

O sistema modelado consiste de um pequeno robô móvel de duas rodas com acionamento diferencial, similar ao utilizado em competições de futebol de robôs. Este robô, de formato cúbico com 7,5 cm de lado, é acionado por dois motores de corrente contínua, um para cada roda, controlado remotamente via rádio-enlace por um computador pessoal.

A figura 3.2 mostra uma representação esquemática do robô modelado. Nesta figura a configuração do robô é representada por sua posição no espaço cartesiano (x e y , que é a posição do centro do robô em relação ao referencial fixo no espaço de trabalho), e pela sua orientação θ (ângulo entre o vetor de orientação do robô e o eixo x do referencial fixo no espaço de trabalho). A posição e a orientação do robô são medidas por processamento da imagem gerada pelo sistema de visão [2].

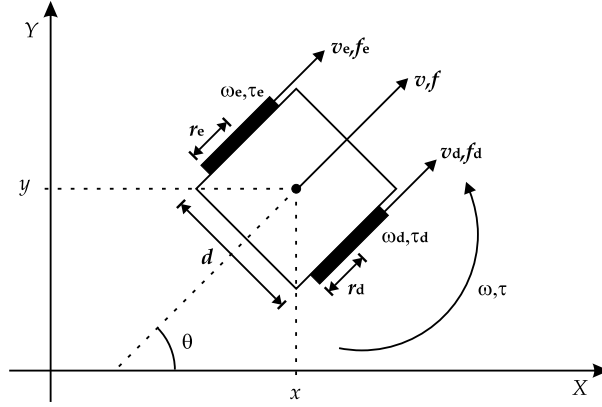


Figura 3.2: Representação esquemática do robô.

A seguinte nomenclatura é adotada:

d é o comprimento do eixo;

$r_{d,e}$ são os raios das rodas direita e esquerda, respectivamente;

$\omega_{d,e}$ são as velocidades angulares das rodas direita e esquerda, respectivamente;

$v_{d,e}, v_{d,e}$ são as velocidades lineares de um ponto nas bordas externas das rodas direita e esquerda, respectivamente;

v é a velocidade linear do robô;

ω é a velocidade angular do robô.

Relação entre Velocidades

As relações entre as velocidades lineares e angulares nas rodas do robô são dadas por:

$$v_d = \omega_d r_d \qquad v_e = \omega_e r_e \qquad (3.1)$$

Como mostra a figura 3.3, para um robô cujo centro de massa se move com velocidade linear v e que gira sobre o centro de massa com velocidade angular ω , os pontos nas bordas externas se moverão com velocidades lineares dadas por:

$$v_d = v + \left(\frac{d}{2}\right) \omega \qquad v_e = v - \left(\frac{d}{2}\right) \omega \qquad (3.2)$$

Das equações (3.1) e (3.2), chega-se a

$$v = \omega_d \frac{r_d}{2} + \omega_e \frac{r_e}{2} \qquad \omega = \omega_d \frac{r_d}{d} - \omega_e \frac{r_e}{d}$$

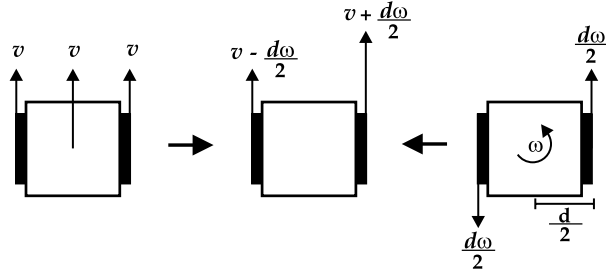


Figura 3.3: Composição de velocidades no robô.

que podem ser representadas na forma matricial como:

$$\mathbf{u} = {}^uT_\omega \boldsymbol{\omega} \quad (3.3)$$

ou, de forma equivalente, por:

$$\boldsymbol{\omega} = ({}^uT_\omega)^{-1} \mathbf{u} \quad (3.4)$$

onde

$$\mathbf{u} = \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad {}^vT_\omega = \begin{bmatrix} r_d/2 & r_e/2 \\ r_d/d & -r_e/d \end{bmatrix},$$

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_d \\ \omega_e \end{bmatrix} \quad \text{e} \quad \omega T_u = ({}^uT_\omega)^{-1} = \begin{bmatrix} 1/r_d & d/2r_d \\ 1/r_e & -d/2r_e \end{bmatrix}$$

O vetor \mathbf{u} representa as velocidades no referencial do robô e $\boldsymbol{\omega}$ é o vetor de velocidades no espaço dos atuadores.

As duas rodas do robô em estudo impedem que ele se movimente lateralmente, tendo liberdade para deslocar-se apenas na direção em que está orientado. Isto é expresso por restrições matemáticas que se enquadram na categoria das restrições não-holonômicas. De acordo com a figura 3.4 para deslocamentos incrementais, tem-se:

$$dx = v dt \cos \theta \rightarrow \dot{x} = v \cos \theta \quad (3.5)$$

$$dy = v dt \sin \theta \rightarrow \dot{y} = v \sin \theta \quad (3.6)$$

$$d\theta = \omega dt \rightarrow \dot{\theta} = \omega \quad (3.7)$$

Igualando-se o valor de v calculado a partir das equações (3.5) e (3.6), chega-se à restrição não-holonômica:

$$\dot{x} \sin \theta = \dot{y} \cos \theta \quad (3.8)$$

As equações (3.5), (3.6) e (3.7) podem ser escritas em forma matricial

$$\dot{\mathbf{q}} = {}^qT_u \mathbf{u} \quad (3.9)$$

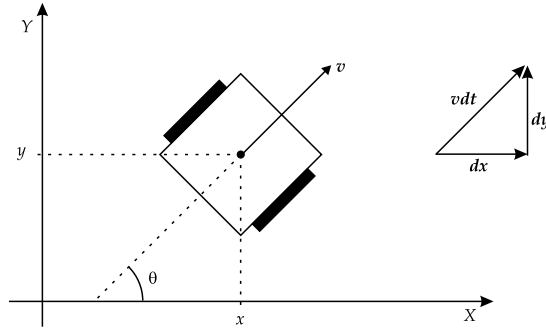


Figura 3.4: Restrições não-holonômicas.

que representa o *modelo cinemático* para este robô, onde

$$\mathbf{q} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \quad \dot{\mathbf{q}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad \text{e} \quad {}^qT_u = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix}$$

O vetor \dot{q} representa as velocidades no espaço de configuração do robô (espaço das variáveis x , y e θ). Substituindo-se a equação (3.3) na equação (3.9) pode-se obter a relação entre as velocidades no espaço de atuadores e no espaço de configuração:

$$\dot{\mathbf{q}} = {}^qT_u {}^uT_\omega \boldsymbol{\omega} \quad (3.10)$$

Pela definição da matriz qT_u , constata-se que:

$${}^qT_u^T {}^qT_u = I$$

onde I é a matriz identidade. Desta forma, pode-se pré-multiplicar a equação (3.9) por ${}^qT_u^T$, o que leva a:

$$\mathbf{u} = {}^qT_u^T \dot{\mathbf{q}} \quad (3.11)$$

Relação entre Esforços nos Diferentes Referenciais

Conforme mostrado na figura 3.2, tem-se que:

f é a força resultante no robô (força no referencial dos eixos principais);

τ é o torque resultante no robô (torque no referencial dos eixos principais);

$f_{d,e}$ são as forças atuantes nas bordas das rodas direita e esquerda, respectivamente (forças no espaço dos atuadores);

$\tau_{d,e}$ são os torques que atuam nas rodas direita e esquerda, respectivamente (torque no espaço dos atuadores).

As forças aplicadas sobre as rodas devidas aos torques desenvolvidos pelos motores são:

$$f_d = \frac{\tau_d}{r_d} \qquad f_e = \frac{\tau_e}{r_e}$$

Assim, de acordo com a figura 3.2, os esforços resultantes são dados por:

$$f = f_d + f_e = \frac{\tau_d}{r_d} + \frac{\tau_e}{r_e} \qquad \tau = \tau_d \frac{d}{2r_d} - \tau_e \frac{d}{2r_e}$$

que na forma matricial podem ser expressos como

$$\mathbf{f}_u = {}^\omega T_u^T \boldsymbol{\tau} \quad (3.12)$$

onde

$$\mathbf{f}_u = \begin{bmatrix} f \\ \tau \end{bmatrix}, \quad {}^\omega T_u = ({}^u T_\omega)^{-1} = \begin{bmatrix} 1/r_d & d/2r_d \\ 1/r_e & -d/2r_e \end{bmatrix} \quad \text{e} \quad \boldsymbol{\tau} = \begin{bmatrix} \tau_d \\ \tau_e \end{bmatrix}$$

O vetor \mathbf{f}_u representa os esforços no referencial dos eixos principais e $\boldsymbol{\tau}$ é o vetor de torques no espaço dos atuadores. A partir da figura 3.2, os esforços resultantes no espaço de configuração podem ser obtidos em função dos esforços no referencial do robô:

$$f_x = f \cos \theta \qquad f_y = f \sin \theta \qquad \tau = \tau$$

Estas relações podem ser representadas de maneira compacta usando notação matricial:

$$\mathbf{f}_q = {}^q T_u \mathbf{f}_u \quad (3.13)$$

onde $\mathbf{f}_q = [f_x \ f_y \ \tau]^T$ é o vetor de esforços no espaço de configuração.

Por fim, substituindo-se a equação (3.12) na equação (3.13) obtém-se a relação entre torques no espaço dos atuadores e esforços no espaço de configuração:

$$\mathbf{f}_q = {}^q T_u {}^\omega T_u^T \boldsymbol{\tau} \quad (3.14)$$

Invertendo a equação (3.14), a relação entre esforços no espaço de configuração e torque no espaço dos atuadores é dada por:

$$\boldsymbol{\tau} = {}^u T_\omega ({}^q T_u)^{-1} \mathbf{f}_q \quad (3.15)$$

Considerando a relação:

$${}^q T_u^T {}^q T_u = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \rightarrow ({}^q T_u)^{-1} = {}^q T_u^T$$

a equação (3.15) pode ser escrita como:

$$\boldsymbol{\tau} = {}^u T_\omega^T {}^q T_u^T \mathbf{f}_q \quad (3.16)$$

3.2.2 Modelagem Dinâmica

Há basicamente duas formas para a obtenção de um modelo dinâmico para um robô móvel. A primeira é a modelagem fenomenológica, que consiste do desenvolvimento matemático a partir das leis físicas envolvidas no processo e na medição de todos os parâmetros que contribuem para a dinâmica do processo (no caso do robô, cerca de 15 parâmetros). Porém, este desenvolvimento nem sempre é trivial, principalmente no que diz respeito à medição de parâmetros. Desta forma, torna-se interessante a adoção de um método baseado nas informações de entrada e saída do sistema, o que caracteriza o segundo método. Este método alternativo é denominado identificação de sistemas, e consiste na estimação dos parâmetros de um determinado sistema a partir de observações de sua resposta perante excitações condicionadas. Há alguns artigos na literatura [16] que fazem uso de técnicas de identificação de sistemas para obtenção de modelos dinâmicos para robôs móveis. A seguir desenvolve-se um modelo dinâmico para o robô da figura 3.2 através da modelagem fenomenológica.

Dinâmica dos Atuadores

As rodas do robô em consideração são acionadas diretamente, sem reduções mecânicas, por motores de corrente contínua. O efeito dinâmico introduzido pelos mesmos pode se tornar relevante em aplicações de controle de trajetórias de alto desempenho.

A dinâmica deste tipo de motor já foi amplamente detalhada na literatura [13]. A figura (3.5) mostra um diagrama do modelo equivalente elétrico da dinâmica dos atuadores do robô móvel, onde os parâmetros dinâmicos são definidos a seguir:

$e_{d,e}$ são as tensões de armadura dos motores direito e esquerdo, respectivamente;

$1/\rho_{d,e}$ são as resistências dos enrolamentos de armadura dos motores;

$i_{d,e}$ são as correntes de armadura dos motores;

$K_{d,e}$ são as constantes de torque dos motores;

$J_{d,e}$ são os momentos de inércia do acoplamento entre o rotor dos motores e as rodas;

$\beta_{d,e}$ são os coeficientes de atrito dos motores.

A equação elétrica de armadura é dada por:

$$\mathbf{i} = \rho \mathbf{e} - \rho \mathbf{K}_m \boldsymbol{\omega} \quad (3.17)$$

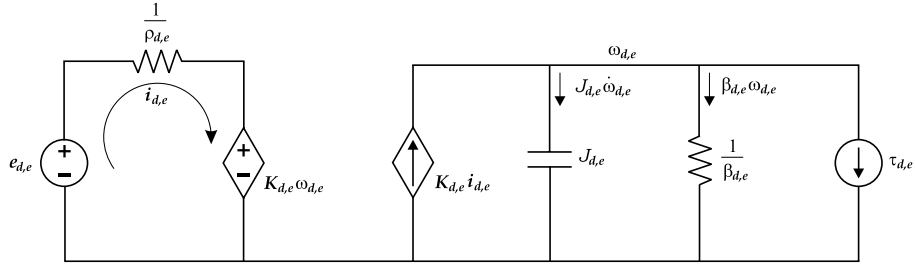


Figura 3.5: Análogo elétrico dos atuadores do robô móvel.

onde

$$\mathbf{e} = \begin{bmatrix} e_d \\ e_e \end{bmatrix}, \quad \boldsymbol{\rho} = \begin{bmatrix} \rho_d & 0 \\ 0 & \rho_e \end{bmatrix},$$

$$\mathbf{i} = \begin{bmatrix} i_d \\ i_e \end{bmatrix} \quad \text{e} \quad \mathbf{K}_m = \begin{bmatrix} K_d & 0 \\ 0 & K_e \end{bmatrix}.$$

A equação eletro-mecânica do motor é dada por:

$$\boldsymbol{\tau} = \mathbf{K}_m \mathbf{i} - \mathbf{J}_m \dot{\boldsymbol{\omega}} - \mathbf{B}_m \boldsymbol{\omega} \quad (3.18)$$

onde o vetor $\boldsymbol{\tau}$ é o mesmo da equação (3.16) e

$$\mathbf{J}_m = \begin{bmatrix} J_d & 0 \\ 0 & J_e \end{bmatrix} \quad \text{e} \quad \mathbf{B}_m = \begin{bmatrix} \beta_d & 0 \\ 0 & \beta_e \end{bmatrix}$$

Substituindo a equação (3.17) na equação (3.18) do motor, obtém-se:

$$\boldsymbol{\tau} = \boldsymbol{\rho} \mathbf{K}_m \mathbf{e} - \mathbf{J}_m \dot{\boldsymbol{\omega}} - (\boldsymbol{\rho} \mathbf{K}_m^2 + \mathbf{B}_m) \boldsymbol{\omega} \quad (3.19)$$

Sendo $\dot{\boldsymbol{\omega}} = {}^\omega T_u \dot{\mathbf{u}}$, tem-se:

$$\boldsymbol{\tau} = \boldsymbol{\rho} \mathbf{K}_m \mathbf{e} - \mathbf{J}_m {}^\omega T_u \dot{\mathbf{u}} - (\boldsymbol{\rho} \mathbf{K}_m^2 + \mathbf{B}_m) {}^\omega T_u \mathbf{u} \quad (3.20)$$

Modelo Dinâmico do Robô

A equação mecânica do robô pode ser obtida através das leis de Newton e Euler do movimento, que podem ser expressas em forma matricial como:

$$\mathbf{f}_u = \mathbf{J}_r \dot{\mathbf{u}} + \mathbf{B}_r \mathbf{u} \quad (3.21)$$

onde

$$\mathbf{J}_r = \begin{bmatrix} m & 0 \\ 0 & J \end{bmatrix} \quad \text{e} \quad \mathbf{B}_r = \begin{bmatrix} \beta_l & 0 \\ 0 & \beta_\theta \end{bmatrix}$$

sendo m a massa do robô, J o seu momento de inércia e $\beta_{l,\theta}$ o coeficiente de atrito viscoso com o solo do robô em movimento linear e rotacional, respectivamente.

Expressando-se \mathbf{f}_u em termos de $\boldsymbol{\tau}$, conforme a equação (3.12), a equação mecânica fica igual a:

$${}^\omega T_u^T \boldsymbol{\tau} = \mathbf{J}_r \dot{\mathbf{u}} + \mathbf{B}_r \mathbf{u} \quad (3.22)$$

Por outro lado, multiplicando a equação (3.20) por ${}^\omega T_u^T$, obtém-se:

$${}^\omega T_u^T \boldsymbol{\tau} = -({}^\omega T_u^T \mathbf{J}_m {}^\omega T_u) \dot{\mathbf{u}} - ({}^\omega T_u^T [\boldsymbol{\rho} \mathbf{K}_m^2 + \mathbf{B}_m] {}^\omega T_u) \mathbf{u} + {}^\omega T_u^T \boldsymbol{\rho} \mathbf{K}_m \mathbf{e} \quad (3.23)$$

Igualando-se as equações (3.22) e (3.23) e isolando-se o termo em \mathbf{e} , chega-se a:

$${}^\omega T_u^T \boldsymbol{\rho} \mathbf{K}_m \mathbf{e} = (\mathbf{J}_r + {}^\omega T_u^T \mathbf{J}_m {}^\omega T_u) \dot{\mathbf{u}} + (\mathbf{B}_r + {}^\omega T_u^T [\boldsymbol{\rho} \mathbf{K}_m^2 + \mathbf{B}_m] {}^\omega T_u) \mathbf{u}$$

que representa o *modelo dinâmico do robô no referencial do robô*. Este modelo pode ser expresso em forma matricial como:

$$K_u \mathbf{e} = M_u \dot{\mathbf{u}} + B_u \mathbf{u} \quad (3.24)$$

onde K_u é um ganho das entradas de controle:

$$K_u = {}^\omega T_u^T \boldsymbol{\rho} \mathbf{K}_m$$

e M_u e B_u são, respectivamente, a matriz de inércia e a matriz de atritos viscosos, ambas constantes, simétricas, definidas positivas e dadas por:

$$\begin{aligned} M_u &= \mathbf{J}_r + {}^\omega T_u^T \mathbf{J}_m {}^\omega T_u \\ B_u &= \mathbf{B}_r + {}^\omega T_u^T [\boldsymbol{\rho} \mathbf{K}_m^2 + \mathbf{B}_m] {}^\omega T_u \end{aligned}$$

Derivando-se a equação 3.11, tem-se que:

$$\dot{\mathbf{u}} = {}^q T_u^T \ddot{\mathbf{q}} + {}^q \dot{T}_u^T \dot{\mathbf{q}}$$

e substituindo-se esta equação na equação (3.24), obtém-se:

$$K_u \mathbf{e} = M_u {}^q T_u^T \ddot{\mathbf{q}} + M_u {}^q \dot{T}_u^T \dot{\mathbf{q}} + B_u {}^q T_u^T \dot{\mathbf{q}} \quad (3.25)$$

Multiplicando ambos os lados da equação (3.25) por ${}^q T_u$, chega-se ao *modelo dinâmico do robô no espaço das configurações*:

$$K_q \mathbf{e} = M_q \ddot{\mathbf{q}} + N_q \dot{\mathbf{q}} + B_q \dot{\mathbf{q}} \quad (3.26)$$

onde

$$\begin{aligned} K_q(\theta) &= {}^q T_u K_u & M_q(\theta) &= {}^q T_u M_u {}^q T_u^T \\ N_q(\theta) &= {}^q T_u M_u {}^q \dot{T}_u^T & B_q(\theta) &= {}^q T_u B_u {}^q T_u^T \end{aligned}$$

sendo que M_q é a matriz de inércia no espaço de configuração e B_q é a matriz de atritos viscosos no espaço das configurações. Ambas são simétricas e definidas positivas. O termo $N_q \dot{\mathbf{q}}$ é o vetor de esforços coriolis-centrífgos. Considerando-se

$${}^q\dot{T}_u^T \dot{\mathbf{q}} = \dot{\theta} \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \dot{\theta} \begin{bmatrix} -\dot{x} \sin \theta + \dot{y} \cos \theta \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

conclui-se que o vetor de esforços coriolis-centrífgos é nulo e, portanto, o modelo dinâmico do robô no espaço das configurações (3.26), que é não-linear e variante no tempo, pode ser reescrito como:

$$K_q \mathbf{e} = M_q \ddot{\mathbf{q}} + B_q \dot{\mathbf{q}} \quad (3.27)$$

Apesar da equação (3.27) representar o modelo dinâmico completo do robô (i.e., incluindo o modelo cinemático), a estratégia de controle foi desenvolvida utilizando-se a representação do modelo dinâmico dada pela equação (3.24) em conjunto com a representação do modelo cinemático dada pela equação (3.9), conforme será apresentado em mais detalhes no capítulo 4. No capítulo 5 também são apresentados resultados de simulações de resposta ao degrau para o modelo aqui obtido. O modelo da equação (3.27) foi integrado ao simulador de sistemas robóticos utilizando os parâmetros dinâmicos apresentados na seção 3.2.2.

Determinação dos Parâmetros do Robô

Para uso em simulações, os valores nominais dos diversos parâmetros físicos do modelo foram levantados para um robô específico, a partir de dados de fabricantes e de ensaios. Um sistema de identificação *online* destes parâmetros deve integrar futuramente o sistema de controle. Os ensaios realizados levaram ao seguinte conjunto de valores, com todas as constantes nas unidades métricas apropriadas, de acordo com o S.I.:

$d = 0,0750 \text{ m}$	
$r_d = 0,0251 \text{ m}$	$r_e = 0,0251 \text{ m}$
$m = 0,352 \text{ kg}$	$J = 13,02 \times 10^{-6} \text{ kg} \cdot \text{m}^2$
$\beta_l = 0,1002 \text{ N} \cdot \text{s} \cdot \text{m}^{-1}$	$\beta_\theta = 0,7 \times 10^{-6} \text{ N} \cdot \text{m} \cdot \text{s} \cdot \text{rad}^{-1}$
$J_d = 3,003 \times 10^{-6} \text{ kg} \cdot \text{m}^2$	$J_e = 3,005 \times 10^{-6} \text{ kg} \cdot \text{m}^2$
$\beta_d = 7,002 \times 10^{-6} \text{ N} \cdot \text{m} \cdot \text{s} \cdot \text{rad}^{-1}$	$\beta_e = 7,002 \times 10^{-6} \text{ N} \cdot \text{m} \cdot \text{s} \cdot \text{rad}^{-1}$
$K_d = 2,04 \times 10^{-2} \text{ V} \cdot \text{s} \cdot \text{rad}^{-1}$	$K_e = 2,01 \times 10^{-2} \text{ V} \cdot \text{s} \cdot \text{rad}^{-1}$
$\rho_d = 0,00577 \text{ } \Omega^{-1}$	$\rho_e = 0,00578 \text{ } \Omega^{-1}$

3.2.3 Aspectos de Não-Holonomia

Conforme apresentado anteriormente, o modelo cinemático restringe as velocidades generalizadas $\dot{\mathbf{q}}$, de forma que estas não podem assumir valores independentes. Esta restrição que aparece no modelo é derivada da estrutura do robô, assumindo a condição de que suas rodas giram sem escorregar no contato com o chão. Isto caracteriza a existência de restrições denominadas não-holonômicas. Para o robô considerado a restrição não-holonômica é dada pela equação:

$$\begin{bmatrix} \sin \theta & -\cos \theta & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = 0 \quad (3.28)$$

que implica que a velocidade linear do centro da roda repousa sobre o plano do corpo da roda, implicando também em uma velocidade lateral nula. Como consequência desta restrição, todas as velocidades generalizadas admissíveis devem pertencer ao espaço nulo do vetor de restrição $C(q) = \begin{bmatrix} \sin \theta & -\cos \theta & 0 \end{bmatrix}$. Desta forma, é possível constatar que a matriz qT_u da equação (3.9) forma, de fato, uma base para o espaço nulo de $C(q)$, pois $C(q){}^qT_u u = 0$, independentemente dos valores de u . Mostrando, assim, que o modelo cinemático representa exatamente as restrições não-holonômicas de movimento para o robô considerado.

3.2.4 Aspectos de Controlabilidade

Antes de prosseguir, é importante comentar a respeito da controlabilidade de sistemas robóticos não-holonômicos. Em Lizarralde, 1998 [10], é abordada uma análise formal e bem detalhada sobre a controlabilidade de sistemas não-lineares (em particular de um uniciclo, que apresenta o mesmo modelo cinemático do robô aqui considerado). Para o escopo deste trabalho, o seguinte resultado desse estudo pode ser destacado: *se um determinado sistema for completamente não-holonômico, ele é também globalmente controlável* [10]. Este resultado garante a controlabilidade do sistema robótico não-holonômico em estudo.

Um outro aspecto importante a respeito das classes de controladores estabilizantes tem como base o Teorema de Brockett [3]. Este teorema enuncia que uma condição necessária para estabilização suave de sistemas regulares sem drift (sistemas que, uma vez zeradas as entradas de controle, permanecem parados em um ponto de equilíbrio) é que o número de entradas seja igual ao número de estados do sistema. Os sistemas robóticos não-holonômicos com rodas violam esta condição, pois o número de entradas é normalmente igual a 2, enquanto o número de estados do sistema é no mínimo igual a 3 (x , y e θ). Sendo assim, para obter um controlador

estabilizante é necessário ou abrir mão da continuidade da lei de controle, ou recorrer a leis de controle variantes no tempo do tipo $u = u(q, t)$. No entanto, será mostrado que, para a estratégia proposta, é utilizado um artifício que permite a utilização de leis de controle suaves e que levam o sistema a um ponto de equilíbrio limite, conforme será apresentado no capítulo 4.

3.3 Simulador de Robôs Móveis

A fim de possibilitar a realização de simulações do sistema robótico apresentado, assim como a realização de testes com diferentes estratégias de controle, foi desenvolvido um simulador em MATLAB para este sistema. O simulador apresenta uma estrutura simples, porém dispõe de recursos para a realização de testes em diversas condições de funcionamento do robô, permitindo desta forma, uma melhor avaliação das estratégias de controle estudadas.

O simulador é um programa escrito na própria linguagem do MATLAB, sem utilização de *toolboxes* exteriores. O programa consiste basicamente de um laço principal que determina quantas iterações de simulação devem ocorrer, a partir de um tempo de simulação fornecido como entrada pelo usuário. A simulação é uma integração numérica, utilizando o método *Runge-Kutta de Quarta Ordem*, da equação diferencial 3.27 que representa o modelo dinâmico do robô móvel.

O simulador é capaz de reproduzir alguns detalhes do sistema de controle real, como a presença de ruídos e incertezas de medição inerentes ao sistema de localização do robô. Uma breve descrição dos dados de entrada para o simulador é feita a seguir.

- **Período de amostragem:** como considera-se que o sistema de controle a ser apresentado será projetado para utilização em um computador, isto é, será um controlador digital, o usuário deve definir o período de amostragem do sistema simulado. Isto fará com que as observações do estado do sistema ocorram nos períodos de amostragem (este é diferente do período de integração utilizado para simulação – definido internamente no próprio MATLAB).
- **Tempo de simulação:** o usuário deve fornecer o tempo de simulação desejado em segundos. Este tempo é dividido pelo período de amostragem, fornecendo desta forma, uma aproximação de quantas iterações serão necessárias para simular durante o tempo especificado pelo usuário.
- **Configuração desejada:** é possível o usuário especificar qual a configuração desejada final para o robô, quando estiver simulando alguma estratégia de controle para estabilização

do sistema.

- **Efeito de pixelização:** este efeito surge do tipo de sistema de localização adotado (visão computacional). Neste tipo de sistema, imagens digitalizadas são obtidas para o processamento e identificação da posição e orientação do robô. Como as imagens são digitalizadas, isto traz incertezas no processamento de localização, visto que haverá uma espécie de quantização na posição obtida (vide figura 3.6). Desta forma, o usuário pode definir se deseja simular com ou sem este efeito, através de um *flag* passado como dado de entrada para o programa. Este efeito é simulado considerando que as imagens obtidas são de 640×480 pixels. Para um maior realismo deste efeito, foram implementados algoritmos de localização bastante similares aos utilizados no sistema de visão para uma melhor reprodução deste efeito.

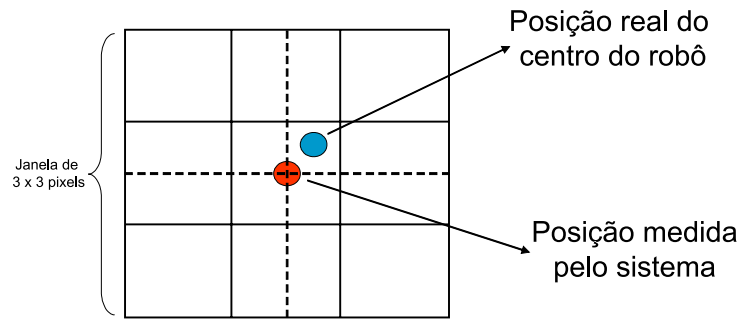


Figura 3.6: Ilustração do erro de quantização gerado pelo sistema de visão.

- **Perturbações:** o usuário pode definir se deseja executar a simulação com ou sem ruídos de processo (através da configuração de *flags* de entrada do programa). Os ruídos introduzidos são ruídos gaussianos de média nula e com baixa variância (podendo ser configurada pelo usuário).
- **Ruído de medição:** além do ruído de processo é possível introduzir na simulação ruídos de medição (através da configuração de *flags* de entrada do programa), que são adicionados às variáveis de saída do sistema (x , y e θ).
- **Quantização do sinal de controle:** como o sistema de controle para o robô móvel normalmente apresenta uma implementação digital, tem-se que o sinal de controle não é contínuo para esses casos. Desta forma, é necessário testar o sistema de controle com os respectivos sinais de comando de forma quantizada, de acordo com a resolução disponível do conversor DA (conversor Digital/Analógico). O usuário pode simular o sistema com ou sem quantização do sinal de controle (através da configuração de um *flag* de entrada do

programa). A resolução atual do simulador para quantização é de 7 bits ($2^7 = 128$ níveis), no entanto, pode ser facilmente alterada pelo usuário.

- **Atraso puro de tempo:** como a plataforma experimental disponível apresenta uma arquitetura onde os sinais de controle são enviados através de um enlace de rádio do computador para os robôs, é interessante poder simular como o sistema de controle se comporta quando os atrasos de transmissão são levados em consideração. Desta forma, o usuário pode especificar, como dado de entrada do programa, o atraso puro de tempo do sistema (em milissegundos).

Conforme apresentado, o simulador inclui os principais efeitos que podem interferir no desempenho do sistema de controle. Este simulador foi utilizado durante o desenvolvimento da estratégia de controle e resultados simulados obtidos através deste serão apresentados nos capítulos que seguem.

Capítulo 4

Estratégia de Controle

O objetivo deste trabalho foi o de investigar estratégias de controle capazes de lidar com o problema de estabilização de robôs móveis, levando em consideração a dinâmica do sistema, e fazendo uso apenas de informações sobre a posição e orientação do robô (obtidas via realimentação visual). Pode-se observar que a maioria das estratégias de controle descritas no capítulo 2 necessitam de informações sobre as velocidades linear e angular do robô, ou até mesmo de suas derivadas. No entanto, nem sempre tais informações são de fácil aquisição e, normalmente, a estimação dessas variáveis de estado não é robusta.

Neste capítulo é apresentada a dedução da estratégia de controle proposta por esta dissertação que atende aos critérios desejados, isto é, uma estratégia capaz de resolver o problema da estabilização dinâmica utilizando apenas informações sobre a posição e orientação do robô. A estratégia foi desenvolvida em duas etapas, a primeira voltada para a solução do problema de controle de posição, e a segunda voltada para solução do problema de orientação.

4.1 Introdução

A idéia principal que suporta a estratégia desenvolvida encontra-se em uma transformação de variáveis envolvidas no modelo dinâmico do robô. Conforme apresentado no capítulo 3, o modelo dinâmico para um robô móvel não-holonômico, acionado por motores DC, pode ser representado pelas equações:

$$\dot{\mathbf{u}} = -M_u^{-1} B_u \mathbf{u} + M_u^{-1} K_u \mathbf{e} \quad (4.1)$$

$$\dot{\mathbf{q}} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \mathbf{u} \quad (4.2)$$

onde $K_u \in \mathbb{R}^{2 \times 2}$ é uma matriz de ganho das entradas de controle $\mathbf{e} = [e_l \ e_r]^T$, $M_u \in \mathbb{R}^{2 \times 2}$ e $B_u \in \mathbb{R}^{2 \times 2}$ são respectivamente a matriz de inércia e a matriz de atrito viscoso, ambas constantes, simétricas e definidas positivas, $\mathbf{q} \in \mathbb{R}^{3 \times 1}$ é o vetor que representa a configuração do robô no espaço cartesiano $(x, y \text{ e } \theta)$, e $\mathbf{u} \in \mathbb{R}^{2 \times 1}$ é o vetor que representa as velocidades linear e angular do robô $(v \text{ e } \omega)$.

As equações (4.1) e (4.2) podem ser re-arranjadas para uma representação no espaço de estado da seguinte forma

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \\ \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -M_v^{-1}B_v \vdots \mathbf{0}_{2 \times 2} \\ \dots\dots\dots \\ G(\theta) \vdots \mathbf{0}_{3 \times 2} \end{bmatrix} \begin{bmatrix} v \\ \omega \\ x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} M_u^{-1}K_u \\ \dots\dots\dots \\ \mathbf{0}_{3 \times 2} \end{bmatrix} \begin{bmatrix} e_l \\ e_r \end{bmatrix} \quad (4.3)$$

onde a matriz $G(\theta)$ representa a matriz de restrições cinemáticas do robô, como segue

$$G(\theta) = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \quad (4.4)$$

As saídas do modelo representado pela equação 4.3 são x , y e θ . Apesar deste modelo incluir a configuração do robô, permitindo o uso de coordenadas cartesianas e ângulos de orientação como referências para o robô, este é um modelo não-linear multivariável e o projeto de controladores baseado neste tipo de modelo não é trivial.

A fim de reduzir a complexidade do modelo, pode-se reescrevê-lo em termos dos deslocamentos linear e angular do robô, aqui denominados de s e θ respectivamente, como segue:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \\ \dot{s} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -M_u^{-1}B_u \vdots \mathbf{0}_{2 \times 2} \\ \dots\dots\dots \\ I_{2 \times 2} \vdots \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} v \\ \omega \\ s \\ \theta \end{bmatrix} + \begin{bmatrix} M_u^{-1}K_u \\ \dots\dots\dots \\ \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} e_l \\ e_r \end{bmatrix} \quad (4.5)$$

Essa mudança de variáveis é possível visto que o deslocamento linear é a integral da velocidade linear e o deslocamento angular é a integral da velocidade angular ($\dot{s} = v$ e $\dot{\theta} = \omega$).

Assim, temos que a representação dada pela equação 4.5 é linear com entradas e_l e e_r e saídas s e θ . Mais adiante, no capítulo 5, será mostrado que é possível ter, através de transformações lineares sobre os sinais de entrada e_l e e_r , entradas e_v e e_ω , onde estas representam excitações para a velocidade linear e angular do sistema, respectivamente. Assim, o modelo da equação 4.5 poderá ser escrito como:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \\ \dot{s} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -M_u^{-1}B_u & \vdots & \mathbf{0}_{2 \times 2} \\ \dots\dots\dots \\ I_{2 \times 2} & \vdots & \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} v \\ \omega \\ s \\ \theta \end{bmatrix} + \begin{bmatrix} P \\ \dots \\ \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} e_v \\ e_\omega \end{bmatrix} \quad (4.6)$$

onde P representa o resultado das transformações lineares sobre as entradas e_l e e_r . Desta forma, é possível projetar um sistema de controle baseado no diagrama de blocos da figura 4.1, se s e θ são mensuráveis e s_{ref} e θ_{ref} são definidos. Este controlador pode ser baseado em qualquer técnica clássica de projeto de sistemas lineares MIMO - *Multiple Inputs Multiple Outputs*, onde o controlador recebe os sinais de erro e gera sinais de entrada para a planta. Considerando o caso no qual o robô é simétrico, isto é, a disposição de seus atuadores é simétrica em relação ao eixo longitudinal do robô e as características destes são idênticas (tornando o sistema desacoplado com relação às entradas), pode-se utilizar controladores SISO - *Single Input Single Output* (PID, por exemplo) para cada malha.

Apesar de aparentemente haver uma solução simples para o problema, esta representação possui um obstáculo para o projeto do controlador: o deslocamento linear (s) através de uma trajetória é praticamente imensurável. A estratégia de controle proposta apresenta uma forma de contornar este problema, conforme será apresentado na seção que segue.

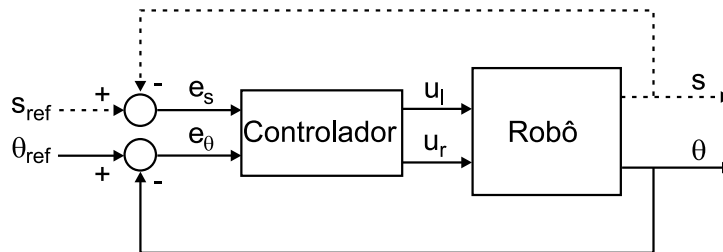


Figura 4.1: Diagrama de blocos do sistema de controle.

A linha tracejada no diagrama de blocos da figura 4.1 representa a impossibilidade de medição direta do deslocamento linear s . O tipo de controlador para o sistema não foi especificado neste diagrama, visto que a estratégia proposta pode ser aplicada com diferentes tipos e implementações de controladores lineares.

4.2 Desenvolvimento da Estratégia de Controle

O problema de estabilização do robô pode ser dividido em dois problemas de controle distintos: um controle de posicionamento e um controle de orientação. O controle de posicionamento deve assegurar que o robô atinja uma determinada posição no espaço cartesiano $(x_{ref}$ e $y_{ref})$, independentemente de sua orientação. O controlador de orientação deve garantir que o robô atinja uma determinada posição $(x_d$ e $y_d)$ com uma orientação desejada (θ_d) . A seguir são apresentadas as transformações de coordenadas do sistema que permitem a derivação das estratégias para controle de posição e de orientação.

4.2.1 Transformação do Modelo Cinemático para Coordenadas Polares

A figura 4.2 representa o problema de controle em coordenadas polares, onde Δl é a distância entre o robô e a posição desejada (x_{ref}, y_{ref}) no espaço cartesiano. Também são definidas duas novas variáveis $\Delta \lambda$ e ϕ . $\Delta \lambda$ representa a distância do robô ao ponto **R**, sendo este o ponto mais próximo da posição desejada e que se encontra sobre a linha de orientação do robô; e ϕ é o ângulo do vetor que conecta o robô à posição desejada. Ainda pode ser definida a variável $\Delta \phi$ como a diferença entre o ângulo ϕ e o ângulo de orientação do robô θ : $\Delta \phi = \phi - \theta$.

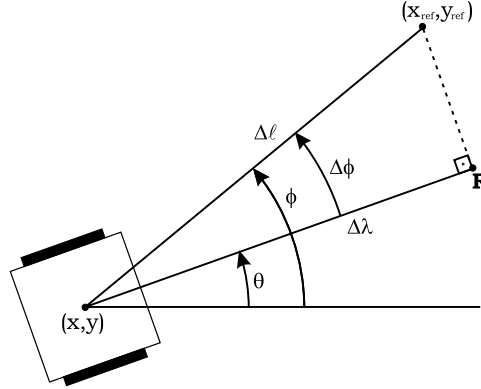


Figura 4.2: Definições envolvidas no problema de controle.

Para esta representação do problema, que aborda a localização do robô em relação ao seu objetivo, é possível derivar o modelo cinemático em coordenadas polares. Tem-se as seguintes definições:

$$\begin{cases} \Delta x = x_{ref} - x \\ \Delta y = y_{ref} - y \end{cases}$$

que também podem ser expressos por

$$\begin{cases} \Delta x = \Delta l \cdot \cos\phi \\ \Delta y = \Delta l \cdot \sin\phi \end{cases}$$

e

$$\begin{cases} \Delta l = \sqrt{(\Delta x)^2 + (\Delta y)^2} \\ \phi = \tan^{-1}\left(\frac{\Delta y}{\Delta x}\right) \end{cases}$$

Assim, é possível derivar Δl e ϕ para a obtenção do modelo, da seguinte forma:

$$\begin{aligned} \dot{\Delta l} &= \frac{1}{2} \cdot \frac{(2 \cdot \Delta x \cdot (-\dot{x}) + 2 \cdot \Delta y \cdot (-\dot{y}))}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} \\ \dot{\Delta l} &= \frac{(-\dot{x} \cdot \Delta x + -\dot{y} \cdot \Delta y)}{\Delta l} \\ \dot{\Delta l} &= -\dot{x} \cdot \cos\phi + -\dot{y} \cdot \sin\phi \end{aligned} \tag{4.7}$$

mas $\dot{x} = v \cdot \cos\theta$ e $\dot{y} = v \cdot \sin\theta$. Substituindo em 4.7 obtém-se:

$$\begin{aligned} \dot{\Delta l} &= -v \cdot (\cos\phi \cdot \cos\theta + \sin\phi \cdot \sin\theta) \\ \dot{\Delta l} &= -v \cdot \cos(\Delta\phi) \end{aligned} \tag{4.8}$$

De forma similar, obtém-se para $\dot{\phi}$:

$$\begin{aligned} \dot{\phi} &= \frac{\left(\frac{-\dot{y} \cdot \Delta x + \dot{x} \cdot \Delta y}{\Delta x^2}\right)}{1 + \left(\frac{\Delta y}{\Delta x}\right)^2} \\ \dot{\phi} &= \frac{(-\dot{y} \cdot \Delta x + \dot{x} \cdot \Delta y)}{(\Delta l)^2} \\ \dot{\phi} &= v \cdot \frac{(-\sin\theta \cdot \Delta y + \cos\theta \cdot \Delta x)}{(\Delta l)^2} \\ \dot{\phi} &= v \cdot \frac{(-\sin\theta \cdot \cos\phi + \cos\theta \cdot \sin\phi)}{\Delta l} \\ \dot{\phi} &= v \cdot \frac{(\sin\phi \cdot \cos\theta - \cos\phi \cdot \sin\theta)}{\Delta l} \\ \dot{\phi} &= v \cdot \frac{\sin(\Delta\phi)}{\Delta l} \end{aligned} \tag{4.9}$$

Desta forma, agrupando os resultados obtidos em 4.8 e 4.9 obtém-se o modelo cinemático em coordenadas polares, dado por:

$$\begin{bmatrix} \dot{\Delta l} \\ \dot{\Delta\phi} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -\cos(\Delta\phi) & 0 \\ \frac{\sin(\Delta\phi)}{\Delta l} & -1 \\ \frac{\sin(\Delta\phi)}{\Delta l} & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{4.10}$$

Uma representação similar a esta é utilizada por Aicardi et al. [1] na derivação da lei de controle POL (Controle por Coordenadas Polares), e é citada em [6] como uma das formas de

representação do modelo cinemático. Assim como citado em Aicardi [1] o modelo dado pela equação (4.10), por estar representado através de coordenadas polares, é válido apenas para valores não nulos da distância Δl , pois quando $\Delta l = 0$ temos que ϕ e $\Delta\phi$ são indefinidos. Isto mostra que não há uma correspondência perfeita com o modelo representado pela equação 4.2, quando $\Delta l = 0$. Baseado neste fato, pode-se dizer que as estratégias de controle a serem apresentadas, apesar de estarem aparentemente não conformes com o Teorema de Brockett (leis de controle contínuas e invariantes no tempo), apresentam uma descontinuidade ao atingir-se objetivo de controle. Isto ocorre porque o modelo de coordenadas polares faz uso da variável ϕ que é calculada a partir da função $\tan^{-1}\left(\frac{y-y_d}{x-x_d}\right)$, que não é definida para a condição $x = x_d$ e $y = y_d$.

Apesar de utilizar a mesma premissa adotada por Aicardi em [1] (transformação de coordenadas para permitir a derivação de leis de controle contínuas e invariantes no tempo), o desenvolvimento da estratégia de controle se dá de forma completamente diferente. Aicardi [1] deriva leis de controle a partir de funções candidatas de Lyapunov, enquanto neste trabalho a estratégia é desenvolvida a partir da mecânica e características de movimento do sistema.

A partir da transformação de variáveis apresentada e das considerações expostas serão apresentadas as estratégias de controle de posição e de orientação desenvolvidas.

4.2.2 Controlador de Posição

Na introdução deste capítulo foi apresentada uma forma para resolver o problema de estabilização através da uma mudança nas variáveis de controle. Esta estratégia é representada através do diagrama de blocos na figura 4.1. Mostrou-se também que há dificuldades envolvidas na estratégia proposta referente à medição do deslocamento linear s e à geração de referências s_{ref} . No entanto, se, ao invés de observarmos as saídas e entradas do sistema, observarmos apenas os erros referentes ao deslocamento linear (e_s , erro linear) e deslocamento angular (e_θ , erro angular), é possível contornar este problema. Desta forma, através da análise do problema de controle de posição é possível definir os erros linear e angular, conforme será apresentado a seguir.

Ao observar a figura 4.2 pode-se verificar que o problema de controle de posição pode ser resolvido se for garantido que $\Delta l \rightarrow 0$ quando $t \rightarrow \infty$. Também é possível concluir que:

$$\Delta l = \frac{\Delta\lambda}{\cos(\Delta\phi)} \quad (4.11)$$

Assim, se $\Delta\lambda \rightarrow 0$ temos que $\Delta l \rightarrow 0$. Para que isto ocorra é necessário desenvolver uma

estratégia de controle capaz de garantir a convergência de $\Delta\lambda$ para zero ¹. A partir desses apontamentos é possível chegar às definições para os erros linear e angular.

Conforme o exposto pelo problema de controle de posição pode-se definir diretamente o erro angular $e_\theta = \Delta\phi$. Para fazer $e_\theta = \Delta\phi$ basta definir $\theta_{ref} = \phi$, obtendo:

$$e_\theta = \theta_{ref} - \theta = \phi - \theta = \Delta\phi \quad (4.12)$$

Para obter isto a partir dos dados fornecidos pelo sistema de localização absoluta é necessário fazer:

$$\theta_{ref} = \tan^{-1} \left(\frac{y_{ref} - y}{x_{ref} - x} \right) \quad (4.13)$$

Discutiu-se anteriormente que o cálculo de e_s não é geralmente muito simples, visto que o sinal de saída s (figura 4.1) não pode ser medido e que não há um método simples e razoável para seu cálculo. No entanto, se for definido o ponto **R** na figura 4.2 como o ponto de referência para o controlador linear, apenas neste caso é verdade que $e_s = s_{ref} - s = \Delta\lambda$. Assim, com base nos dados disponíveis do sistema de localização absoluta, é possível calcular:

$$e_s = \Delta\lambda = \Delta l \cdot \cos(\Delta\phi) = \sqrt{(x_{ref} - x)^2 + (y_{ref} - y)^2} \cdot \cos \left[\tan^{-1} \left(\frac{y_{ref} - y}{x_{ref} - x} \right) - \theta \right] \quad (4.14)$$

Considerando as definições dos erros, e_θ e e_s ,

$$\begin{cases} e_\theta = \Delta\phi \\ e_s = \Delta l \cdot \cos(\Delta\phi) \end{cases} \quad (4.15)$$

juntamente com a estratégia de controle apresentada pelo diagrama de blocos da figura 4.1, basta projetar um controlador que garanta a convergência do erro e_s para zero para que seja atingida a solução do problema de controle de posição (para garantir que o $\cos(e_\theta) \neq 0$ é suficiente implementar um controlador para fazer e_θ convergir para zero). Pode-se também construir um novo diagrama de blocos de controle com as definições expostas, conforme figura 4.3. Deve-se lembrar que esta estratégia permite a utilização de diversos tipos de controladores, ficando este a critério do projetista.

No capítulo 6 são apresentados e discutidos resultados de experimentos e simulações obtidos através da aplicação da estratégia proposta para controle de posição.

Modificação da estratégia de controle de posição

Da forma como foram definidos os erros para este sistema, o robô sempre atinge o objetivo de controle orientado para frente, isto é, caso o objetivo encontre-se atrás do robô (se o robô possuir

¹Para o controle de posição não é necessária a convergência de $\Delta\phi$ para zero, deve-se apenas garantir que $\cos(\Delta\phi) \neq 0$.

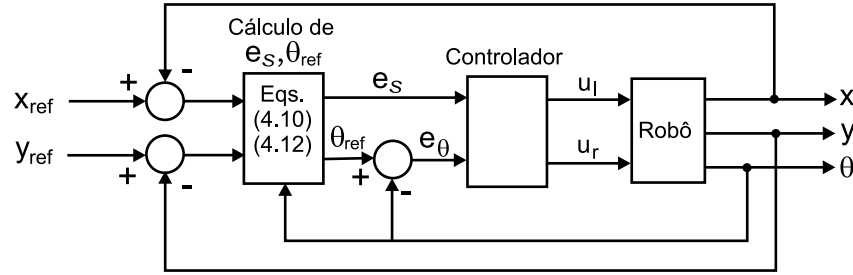


Figura 4.3: Diagrama de blocos da estratégia de controle de posição.

parte frontal e traseira, ou se esta definição existir), este se voltará para o alvo tentando atingí-lo frontalmente. Para contornar este problema, e buscando uma aproximação mais inteligente do objetivo, é possível introduzir uma condição de verificação para modificar o erro angular, conforme a expressão abaixo:

$$\begin{cases} e_\theta = \Delta\phi & , \quad |\Delta\phi| < \pi/2 \\ e_\theta = \Delta\phi - \pi \cdot \text{sgn}(\Delta\phi) & , \quad |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.16)$$

Apesar da modificação no erro angular proposta pela equação 4.16, o erro linear não deverá sofrer modificações. No capítulo 6 serão apresentados resultados de simulação envolvendo a modificação aqui proposta e sua comparação com a estratégia original.

4.2.3 Controlador de Orientação

O problema de controle de orientação é similar ao problema de controle de posição, porém, além de garantir a estabilização em um determinado ponto do espaço cartesiano, é necessário garantir que a orientação final (θ) do robô seja igual à orientação desejada (θ_d). A figura 4.4 apresenta a possibilidade de atingir a estabilização dinâmica do robô a partir de um gerador de trajetórias combinado à estratégia de controle de posição proposta, o que requer cálculos complexos e, conseqüentemente, *hardware* adequado. No entanto, é possível criar um sistema

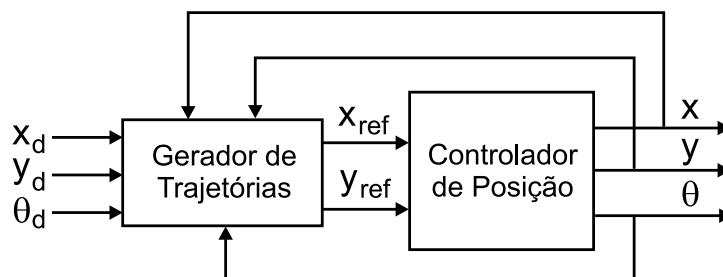


Figura 4.4: Diagrama de blocos da estratégia de controle de posição combinada a um gerador de trajetórias.

similar a um gerador de trajetórias que, a partir da posição e orientação do robô, calcula qual

a nova referência para o robô de forma que este se aproxime, a cada passo, do objetivo de controle. Este sistema pode ser denominado de sistema de referência móvel ou pseudo-gerador de trajetórias. O termo pseudo-gerador de trajetórias é utilizado pois não é realizado o cálculo completo da trajetória que o robô deve seguir (funções $x_{ref}(t)$ e $y_{ref}(t)$ calculadas com base na dinâmica e restrições não-holonômicas do sistema) para atingir o objetivo de controle, mas uma nova referência ($x_{ref}(t)$ e $y_{ref}(t)$) é gerada a cada passo de controle de forma a permitir a aproximação do objetivo de controle com a orientação desejada. Assim, não se tem controle sobre a trajetória que o robô executará para atingir o objetivo, apenas garante-se que este o atingirá. A figura 4.5 apresenta a idéia do esquema de referência móvel e deve-se reforçar que este sistema **não é um gerador de trajetórias** e, desta forma, não inclui restrições relativas à dinâmica do sistema. Assim, apesar de haver a possibilidade da utilização desta técnica, os argumentos apontados como suficientes para o atingimento do objetivo de controle de posição (projetar controladores para levar os erros do sistema para zero), passam a ser necessários porém não suficientes, conforme será abordado no capítulo 5. O desenvolvimento deste sistema é apresentado a seguir.

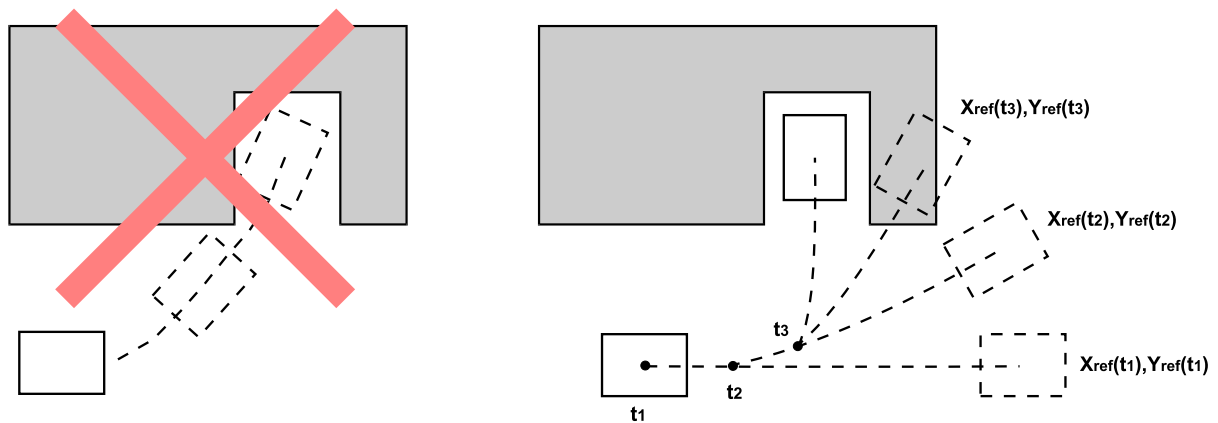


Figura 4.5: Idéia do controle de orientação baseado em referência móvel.

Primeiramente é necessário analisar a característica do problema de controle de orientação. Observando a figura 4.6 pode-se perceber que agora são definidas novas variáveis que envolvem a orientação de chegada do robô (θ_d). A variável γ representa a diferença entre o ângulo ϕ e a orientação desejada para o robô (θ_d): $\gamma = \phi - \theta_d$.

O objetivo do controle de orientação é fazer $\theta = \theta_d$ e $\Delta l \rightarrow 0$ quando $t \rightarrow \infty$. Isto pode ser atingido garantindo $\theta \rightarrow \phi$ e $\phi \rightarrow \theta_d$, juntamente com $\Delta l \rightarrow 0$ quando $t \rightarrow \infty$.

A partir das definições expostas pretende-se utilizar a arquitetura montada para o controle de posição, com algumas modificações, em conjunto com um sistema de referência móvel. Para tal, antes de derivar as equações deste sistema, será necessário definir os erros desejados para o

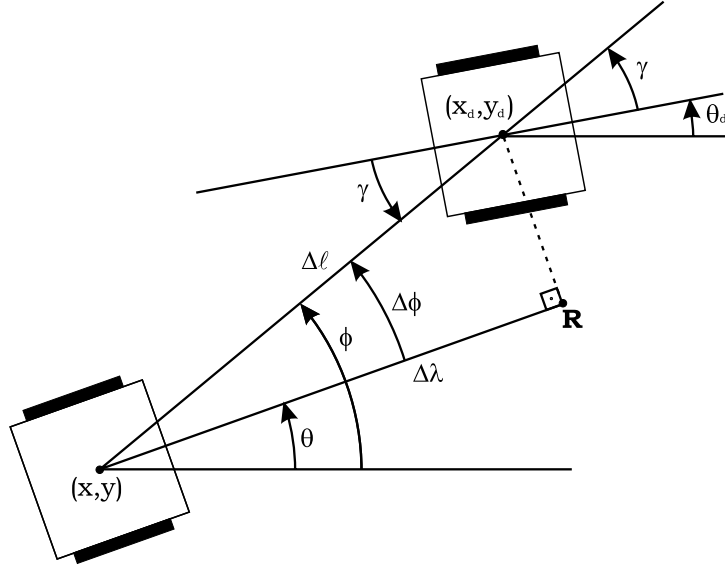


Figura 4.6: Definições envolvidas no problema de estabilização.

problema de orientação e em seguida transformá-los em referências móveis para o controlador de posição. Este processo inclui basicamente a redefinição do erro angular.

No caso do controle de posição foi definido $e_\theta = \Delta\phi$ de forma que, para $e_\theta = 0$, $\phi = \theta$. Para atingir o objetivo do controle de orientação, é necessário fazer $\phi = \theta = \theta_d$. Assim, é possível definir a seguinte expressão para o erro angular:

$$\begin{aligned} e_{\theta_{orient}} &= (\phi - \theta) + (\phi - \theta_d) \\ e_{\theta_{orient}} &= \Delta\phi + \gamma \end{aligned} \quad (4.17)$$

Após a definição do erro angular ($e_{\theta_{orient}}$) é possível definir também o erro linear ($e_{s_{orient}}$), onde o subscrito *orient* denota apenas a diferença das definições de erro para o controle de orientação. O erro linear para o problema de orientação é definido como o ponto mais próximo, que encontra-se sobre a reta de direção do robô, da referência de posição gerada pelo erro angular ($e_{\theta_{orient}}$) e distante Δl da posição atual do robô. Assim, podem ser definidos os seguintes erros para o problema de controle de orientação:

$$\begin{cases} e_{\theta_{orient}} = \Delta\phi + \gamma \\ e_{s_{orient}} = \Delta l \cdot \cos(\Delta\phi + \gamma) \end{cases} \quad (4.18)$$

Agora é possível transformar os erros definidos por 4.18 em um sistema de referência móvel. As expressões utilizadas para a geração de uma referência móvel são dadas a seguir:

$$\begin{cases} x_{ref} = x + \Delta l \cdot \cos(\phi + \gamma) \\ y_{ref} = y + \Delta l \cdot \sin(\phi + \gamma) \end{cases} \quad (4.19)$$

A partir dessas equações é possível chegar às equações para o erro linear e angular para o controle de orientação. O desenvolvimento a seguir apenas ilustra como a equação 4.19 é utilizada para de forma reversa para obtenção dos erros definidos.

$$\begin{aligned}\sqrt{(x_{ref} - x)^2 + (y_{ref} - y)^2} &= \sqrt{\Delta l^2 \cdot \cos^2(\phi + \gamma) + \Delta l^2 \cdot \sin^2(\phi + \gamma)} \\ &= \sqrt{\Delta l^2 \cdot (\cos^2(\phi + \gamma) + \sin^2(\phi + \gamma))} \\ &= \Delta l\end{aligned}$$

e

$$\begin{aligned}\tan^{-1}\left(\frac{y_{ref} - y}{x_{ref} - x}\right) &= \tan^{-1}\left(\frac{\Delta l \cdot \sin(\phi + \gamma)}{\Delta l \cdot \cos(\phi + \gamma)}\right) \\ &= \tan^{-1}\left(\frac{\sin(\phi + \gamma)}{\cos(\phi + \gamma)}\right) \\ &= \tan^{-1}(\tan(\phi + \gamma)) \\ &= \phi + \gamma\end{aligned}$$

Assim, aplicando-se a estratégia de controle de posição (onde $e_{\theta_{pos}}$ e $e_{s_{pos}}$ denotam os erros para esta estratégia) juntamente com as referências geradas por 4.19 obtém-se:

$$\begin{cases} e_{\theta_{pos}} = (\phi + \gamma) - \theta \\ e_{s_{pos}} = \Delta l \cdot \cos[(\phi + \gamma) - \theta] \end{cases}$$

Como $\Delta\phi = \phi - \theta$, tem-se que $e_{\theta_{pos}}$ e $e_{s_{pos}}$ são exatamente iguais aos erros definidos para o controle de orientação, $e_{\theta_{orient}}$ e $e_{s_{orient}}$, conforme equação (4.18).

A partir dessas definições é apresentada uma seqüência de figuras, a fim de permitir um melhor entendimento do funcionamento desta estratégia e da evolução da referência móvel. Na figura 4.7 apresenta-se a condição inicial do problema, onde a configuração do robô é $q = \begin{bmatrix} 0 & 0 & \pi/2 \end{bmatrix}^T$ e a configuração desejada é $q = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^T$. No primeiro momento tem-se $e_{\theta_{orient}} = 0$ e $e_{s_{orient}} = \Delta l$ e o robô tenderá a realizar o movimento puramente para frente. Na figura 4.8 pode-se observar que há alterações nas variáveis $\Delta\phi$, γ e Δl de forma que a referência móvel assume uma nova posição, aparecendo agora o efeito do erro angular. Seguindo o mesmo raciocínio, pode-se observar na seqüência de figuras (4.7 a 4.11) que as variáveis $\Delta\phi$, γ e Δl vão se modificando e a referência móvel se ajusta de forma que os erros angular e linear com relação à configuração desejada tendem para zero. Na figura 4.11 tem-se $\gamma = 0$ e o problema se resume ao problema de controle de posição. A figura 4.12 apresenta a ação do controle de posição a partir da referência móvel gerada para o robô, onde os erros do controle de posição são destacados em vermelho e $\Delta\phi_{pos}$ denota a perspectiva de $\Delta\phi$ observada pelo controlador de posição. Para cada nova referência o controlador atua conforme esquema da figura 4.12.

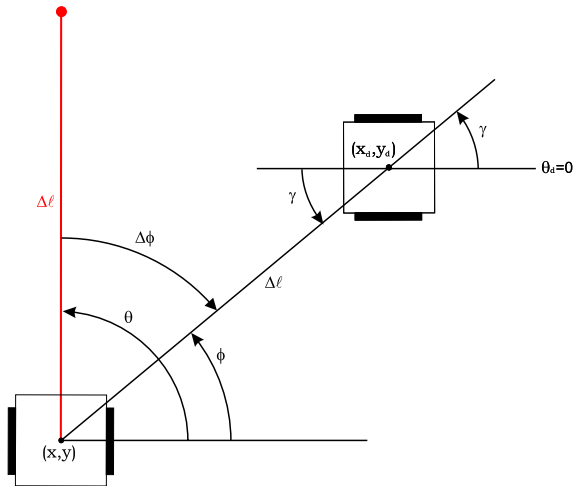


Figura 4.7: Exemplo do controle de orientação (a).

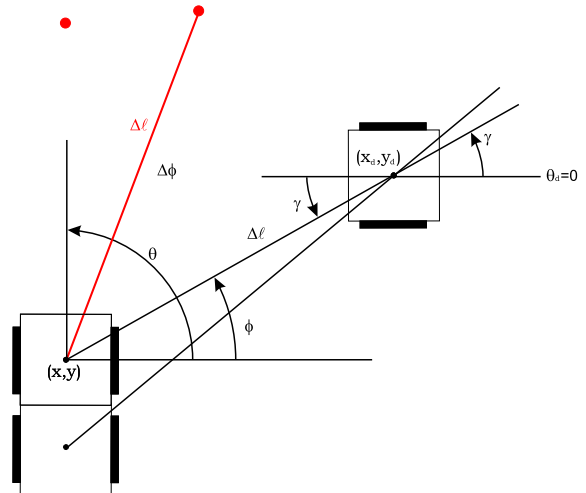


Figura 4.8: Exemplo do controle de orientação (b).

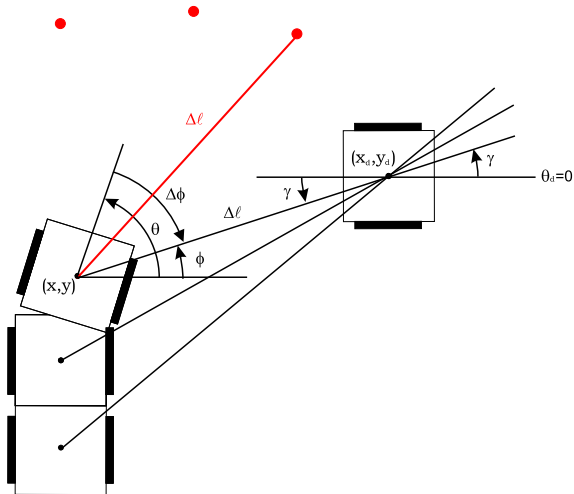


Figura 4.9: Exemplo do controle de orientação (c).

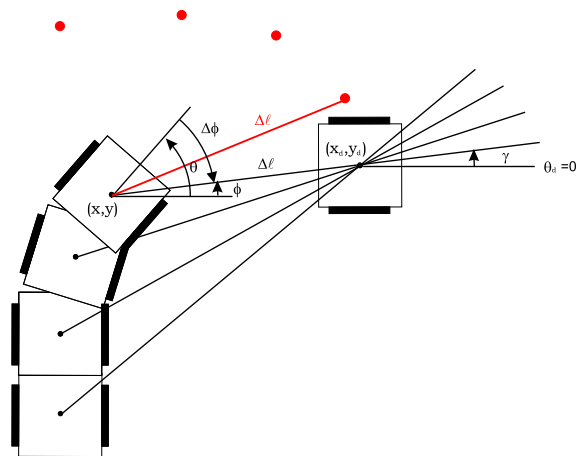


Figura 4.10: Exemplo do controle de orientação (d).

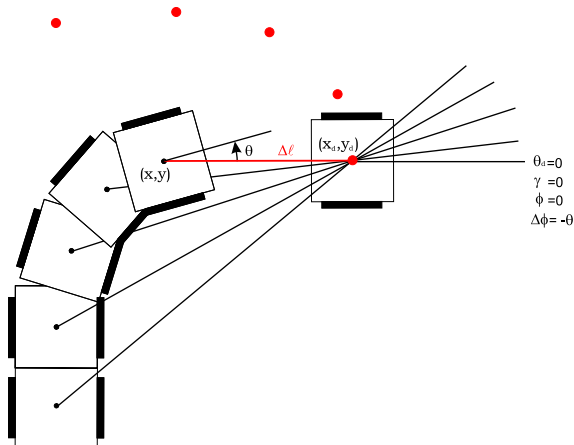


Figura 4.11: Exemplo do controle de orientação (e).

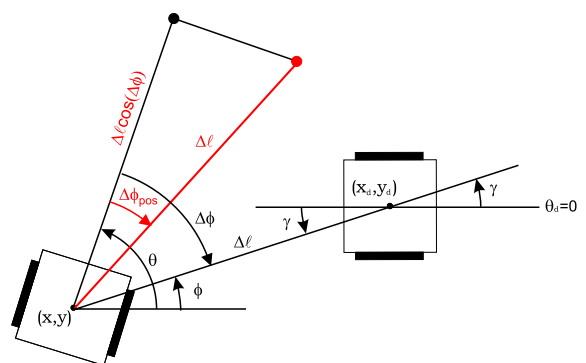


Figura 4.12: Exemplo de referência móvel em conjunto com controle de posição.

As definições apresentadas para o controle de orientação, em alguns casos, não atendem ao objetivo de controle. Isto ocorre principalmente quando a dinâmica do sistema (relacionada ao movimento linear) apresenta sobre-sinal, ou quando a posição desejada encontra-se atrás do robô. A seguir são apresentados os aspectos que contribuem para este fato.

Considerando a definição de erro angular, tem-se, para $e_\theta = 0$:

$$\begin{aligned} 0 &= \Delta\phi + \gamma \\ 0 &= \phi - \theta + \phi - \theta_d \\ 0 &= 2 \cdot \phi - \theta - \theta_d \\ \phi &= \frac{\theta + \theta_d}{2} \end{aligned}$$

de onde pode-se observar que, apesar de existir a solução desejada $\phi = \theta = \theta_d$, há mais de uma solução para ϕ (com θ_d constante) que atende à equação $e_\theta = 0$. Assim, a utilização dos erros definidos pela equação 4.18 não é suficiente para o atingimento do objetivo de controle (principalmente quando a resposta do movimento linear apresenta sobre-sinal). A figura 4.13 apresenta uma situação onde a estratégia (equação 4.18) não funciona (considerando $\theta_d = \pi/4$). Observando esta figura, nota-se que no ponto 1 o robô está se aproximando do objetivo com

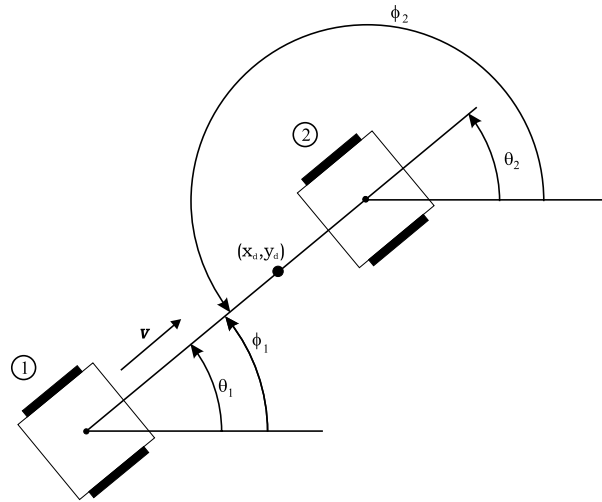


Figura 4.13: Exemplo onde os erros definidos não atendem ao problema de controle.

$\phi = \theta = \theta_d$, ou seja, $e_\theta = \Delta\phi + \gamma = 0$ e $e_s = \Delta l \cdot \cos(\Delta\phi + \gamma) = \Delta l$. Devido à dinâmica do sistema, o robô passa por seu objetivo chegando ao ponto 2, onde tem-se $e_\theta = 2\pi$ (ou equivalentemente $e_\theta = 0$) e $e_s = \Delta l \cdot \cos(\Delta\phi + \gamma)$ (ou equivalentemente $e_s = \Delta l$). Como o erro angular permanece zero e o erro linear permanece igual a Δl a tendência de evolução do sistema será o afastamento do ponto desejado com $\Delta l \rightarrow \infty$. Nota-se que no ponto 2, apesar de $e_\theta = 0$, tem-se $\Delta\phi \neq 0$ e $\gamma \neq 0$. Assim fica clara a necessidade de um método capaz de fazer com que

o robô “perceba” a necessidade de retornar ao objetivo, sem alterar o erro angular, de forma a garantir a estabilização.

O método utilizado para resolver o problema exposto permite ao robô identificar onde se encontra o objetivo de controle em relação à sua orientação. No caso em que o objetivo encontrar-se atrás do robô (vide figura 4.14) então haverá uma alteração de sinal em seu erro linear de forma a permitir o movimento contrário e, caso negativo, não há alteração nos erros definidos pela equação 4.18. A expressão para o erro linear que incorpora o método citado é dada por:

$$\begin{cases} e_{s_{orient}} = \Delta l \cdot \cos(\Delta\phi + \gamma) & , |\Delta\phi| < \pi/2 \\ e_{s_{orient}} = \Delta l \cdot \cos(\Delta\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) & , |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.20)$$

Isto é, o método consiste em subtrair ou somar π a $(\Delta\phi + \gamma)$, de acordo com o seu sinal, quando for detectado que $\Delta\phi \geq \pi/2$ (objetivo encontra-se atrás do robô). De forma similar a expressão para o sistema de referência móvel deve ser reescrita, da seguinte forma:

$$\begin{cases} x_{ref} = x + \Delta l \cdot \cos(\phi + \gamma) \\ y_{ref} = y + \Delta l \cdot \text{sen}(\phi + \gamma) \end{cases} , |\Delta\phi| < \pi/2 \\ \begin{cases} x_{ref} = x + \Delta l \cdot \cos(\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) \\ y_{ref} = y + \Delta l \cdot \text{sen}(\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) \end{cases} , |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.21)$$

Utilizando as expressões dadas por 4.21 é necessário proceder correção no erro angular quando $|\Delta\phi| \geq \pi/2$, visto que este deverá permanecer sempre o mesmo, independente da orientação do robô em relação ao seu objetivo. Assim, tem-se para o erro angular do controlador de posição:

$$\begin{cases} e_{\theta_{pos}} = \Delta\phi_{pos} & , |\Delta\phi| < \pi/2 \\ e_{\theta_{pos}} = \Delta\phi_{pos} + \pi \cdot \text{sgn}(\Delta\phi_{pos}) & , |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.22)$$

onde $\Delta\phi_{pos}$ é o erro angular enxergado pelo controlador de posição em relação à referência móvel (x_{ref} e y_{ref} - ver figura 4.12) e $\Delta\phi$ é o erro angular relativo ao objetivo de controle (x_d e y_d). Assim, retira-se a influência no erro angular gerada pela expressão 4.21, na condição $|\Delta\phi| \geq \pi/2$. Esta correção do erro angular, dada por 4.22, é o único ponto que impede a aplicação direta do sistema de referência móvel em conjunto com a estratégia de controle de posição. No entanto, basta introduzir esta modificação para permitir a utilização do sistema de referência móvel, conforme figura 4.4.

O método proposto introduz um chaveamento na lei de controle, porém isto permite o atingimento do objetivo de forma mais natural. Com este método evita-se que o robô se aproxime do objetivo sempre “de frente”. A figura 4.14 apresenta um caso no qual o robô já se encontra

orientado, sendo necessário apenas um movimento para trás para que o objetivo de controle seja atingido. No entanto, isto só se torna possível com o método introduzido para correção do erro linear. Pois, para valores de $\Delta\phi \geq \pi/2$, tem-se que o alvo encontra-se “atrás” do robô e, desta forma, o erro linear sofre uma correção de π radianos, permitindo que o robô evolua no sentido contrário ao de sua orientação.

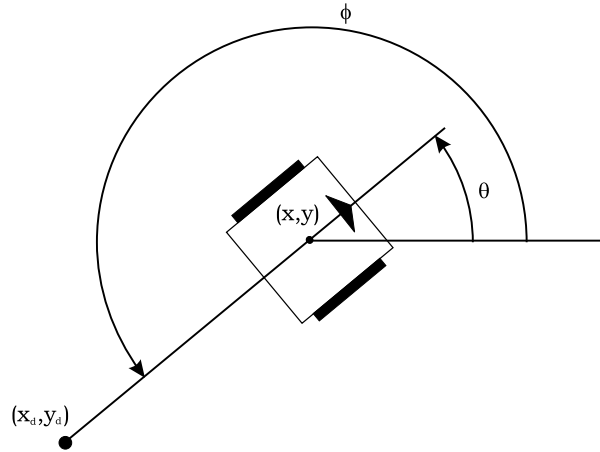


Figura 4.14: Exemplo onde o objetivo x_d e y_d encontram-se atrás do robô.

É importante lembrar que todas as características e definições apresentadas não abordam, neste nível, aspectos dinâmicos do robô, mas apenas as características cinemáticas do problema de controle de orientação. Diferentemente do controle de posição, a estratégia de orientação é mais complexa, conforme descrito no início desta seção. Mais adiante, no capítulo 5, serão abordados aspectos de estabilidade referentes à estratégia de controle de orientação.

Apesar de ter sido derivada a expressão para o sistema de referência móvel, este não precisa ser utilizado para implementação da estratégia, isto é, pode-se empregar diretamente as definições de erro geradas para o problema de controle de orientação. Contudo, com o sistema de referência móvel, é possível ter uma melhor visualização do efeito do controle de orientação, conforme será apresentado no capítulo 6 de resultados.

No capítulo 6 serão apresentados e discutidos resultados simulados e experimentais envolvendo diversas configurações iniciais e finais para o robô.

4.3 Resumo das Estratégias de Controle

Aqui é apresentado um resumo das estratégias de controle de posição e orientação propostas, com suas variações.

- Estratégia para Controle de Posição:

$$\begin{cases} e_{s_{pos}} = \Delta l \cdot \cos(\Delta\phi) \\ e_{\theta_{pos}} = \Delta\phi \end{cases} \quad (4.23)$$

- Estratégia para Controle de Posição Modificada:

$$\begin{cases} e_{s_{pos}} = \Delta l \cdot \cos(\Delta\phi) \\ e_{\theta_{pos}} = \Delta\phi & , \quad |\Delta\phi| < \pi/2 \\ e_{\theta_{pos}} = \Delta\phi - \pi \cdot \text{sgn}(\Delta\phi) & , \quad |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.24)$$

- Estratégia para Controle de Orientação por Referência Móvel:

$$\begin{cases} \begin{cases} x_{ref} = x + \Delta l \cdot \cos(\phi + \gamma) \\ y_{ref} = y + \Delta l \cdot \text{sen}(\phi + \gamma) \end{cases} & , \quad |\Delta\phi| < \pi/2 \\ \begin{cases} x_{ref} = x + \Delta l \cdot \cos(\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) \\ y_{ref} = y + \Delta l \cdot \text{sen}(\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) \end{cases} & , \quad |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.25)$$

Com modificação na estratégia de controle de posição dada por:

$$\begin{cases} e_{\theta_{pos}} = \Delta\phi_{pos} & , \quad |\Delta\phi| < \pi/2 \\ e_{\theta_{pos}} = \Delta\phi_{pos} + \pi \cdot \text{sgn}(\Delta\phi_{pos}) & , \quad |\Delta\phi| \geq \pi/2 \end{cases} \quad (4.26)$$

- Estratégia para Controle de Orientação por Definição Direta de Erros:

$$\begin{cases} e_{s_{orient}} = \Delta l \cdot \cos(\Delta\phi + \gamma) & , \quad |\Delta\phi| < \pi/2 \\ e_{s_{orient}} = \Delta l \cdot \cos(\Delta\phi + \gamma - \pi \cdot \text{sgn}(\Delta\phi + \gamma)) & , \quad |\Delta\phi| \geq \pi/2 \\ e_{\theta_{orient}} = \Delta\phi + \gamma \end{cases} \quad (4.27)$$

4.4 Conclusão

Neste capítulo foi apresentado o desenvolvimento das estratégias de controle de posição e orientação, utilizando apenas as informações disponíveis pelo sistema de posicionamento absoluto, conforme proposta inicial. O desenvolvimento de uma estratégia baseada apenas na definição dos erros do sistema tornou-se possível a partir do novo arranjo das variáveis de estado do sistema robótico, tanto no que diz respeito ao modelo dinâmico quanto ao cinemático.

Conforme discutido neste capítulo, a aplicação da estratégia de controle de orientação exige que sejam levados em consideração alguns aspectos (abordados no capítulo 5) relativos à dinâmica dos movimentos do robô no projeto dos controladores.

No capítulo 5 será mostrado que, além de eficácia, as estratégias propostas possuem razoável facilidade para implementação e flexibilidade, no que diz respeito ao tipo de controlador a ser adotado.

Apesar de terem sido introduzidas condições que levam ao chaveamento dos erros em ambas as estratégias, estes não apresentam impactos significativos no sinal de controle de forma a prejudicar o desempenho do sistema, conforme será apresentado no capítulo 6.

Capítulo 5

Projeto e Implementação

As estratégias apresentadas neste trabalho permitem flexibilidade no que diz respeito à escolha do tipo de controlador a ser utilizado. Dependendo das características reais do sistema, controladores mais sofisticados poderão ser exigidos, o que requer um melhor conhecimento nesta área. Isto pode ser necessário, por exemplo, quando o sistema apresentar um forte grau de acoplamento entre os estados que representam a velocidade linear e angular do robô, devido principalmente a diferenças significativas em seus atuadores ou geometria. No entanto, para grande parte dos casos, onde os robôs com acionamento diferencial apresentam atuadores com características muito similares, controladores tipo PID são utilizados, devido à robustez e facilidade de implementação e sintonia. Assim, neste capítulo será abordado o projeto de controladores PID para o sistema, a partir de seu modelo dinâmico. Ainda neste capítulo serão abordados alguns aspectos referentes à estabilidade do sistema quando utilizadas essas estratégias.

5.1 Características do Sistema

No capítulo 3 foram derivados os modelos cinemático e dinâmico para o robô. No entanto, para o desenvolvimento das estratégias de controle de posição e orientação, foi proposta uma transformação de variáveis levando à representação:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \\ \dot{s} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -M_u^{-1}B_u & \vdots & \mathbf{0}_{2 \times 2} \\ \dots\dots\dots\dots\dots\dots \\ I_{2 \times 2} & \vdots & \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} v \\ \omega \\ s \\ \theta \end{bmatrix} + \begin{bmatrix} M_u^{-1}K_u \\ \dots\dots\dots \\ \mathbf{0}_{2 \times 2} \end{bmatrix} \begin{bmatrix} e_l \\ e_r \end{bmatrix} \quad (5.1)$$

No entanto, deseja-se obter entradas de controle e_v e e_ω para a velocidade linear e angular, respectivamente. No capítulo 4 comentou-se sobre a existência de uma transformação linear

sobre as entradas e_l e e_r de forma a obter e_v e e_ω . Devido ao acoplamento natural deste tipo de sistema, gerado por pequenas diferenças nos parâmetros (dinâmicos e geométricos) dos atuadores, não é possível transformar, diretamente (i.e., apenas por uma transformação linear), as entradas e_l e e_r em entradas e_v e e_ω , que representam excitações para a velocidade linear (v) e para a velocidade angular (ω) respectivamente. No entanto, é possível mostrar que, para a condição de regime permanente, isto é possível e há uma transformação linear sobre e_l e e_r tal que:

$$\begin{bmatrix} e_v \\ e_\omega \end{bmatrix} = T \begin{bmatrix} e_l \\ e_r \end{bmatrix} \quad (5.2)$$

onde T pode ser calculado conforme segue. Considerando o modelo representado por 4.5 e tomando as matrizes $A = -M_u^{-1}B_u$ e $B = M_u^{-1}K_u$, tem-se:

$$\dot{\mathbf{v}} = A\mathbf{v} + B\mathbf{e}$$

com $\mathbf{v} = [v \ \omega]^T$ e $\mathbf{e} = [e_l \ e_r]^T$. Assim, para a condição de regime permanente (i.e., $\dot{\mathbf{v}} = 0$), temos:

$$0 = A\mathbf{v} + B\mathbf{e}$$

$$-A\mathbf{v} = B\mathbf{e}$$

$$\mathbf{v} = -A^{-1}B\mathbf{e}$$

Desta forma, pode-se concluir que a matriz T (equação 5.2) é dada por:

$$T = -A^{-1}B \quad (5.3)$$

Com isto, mesmo que o sistema apresente dinâmica acoplada, isto é, mesmo que a matriz A não seja diagonal, tem-se o desacoplamento em regime.

Caso haja acoplamento dinâmico severo é necessário introduzir um controlador capaz de transformar a dinâmica do sistema tornando-o desacoplado. Uma opção é utilizar a técnica *model matching*, através da qual é possível casar a dinâmica do robô com a dinâmica desejada. A vantagem da utilização desta técnica de controle é a possibilidade de lidar com sistemas multivariáveis. No entanto, a técnica de controle a ser implementada não é o objetivo principal deste trabalho e, portanto, não serão aprofundados esses conceitos.

Na seção 3.2.2 do capítulo 3 foram apresentados parâmetros para um sistema robótico obtidos através de ensaios e dados de fabricantes. Com os parâmetros apresentados é possível calcular as matrizes $A = -M_u^{-1}B_u$ e $B = M_u^{-1}K_u$ que descrevem o comportamento dinâmico dos atuadores. Desta forma, aplicando as equações para derivação de A e B obtém-se:

$$A = \begin{bmatrix} -1,1046 & 0,0000 \\ 0,0000 & -1,4824 \end{bmatrix}, \quad B = \begin{bmatrix} 0,1302 & 0,1302 \\ 2,5168 & -2,5168 \end{bmatrix}$$

Assim, a equação dinâmica para os atuadores é dada por:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -1,1046 & 0,0000 \\ 0,0000 & -1,4824 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} + \begin{bmatrix} 0,1302 & 0,1302 \\ 2,5168 & -2,5168 \end{bmatrix} \begin{bmatrix} e_l \\ e_r \end{bmatrix}$$

Neste caso foram utilizados parâmetros idênticos para ambos os atuadores. Por esta razão a matriz A é diagonal e o sistema possui dinâmica desacoplada. No entanto, pode-se ainda realizar a transformação das entradas de controle para e_v e e_ω . Para tal deve-se calcular T , conforme segue:

$$T = -A^{-1}B$$

$$T = - \begin{bmatrix} -1,1046 & 0,0000 \\ 0,0000 & -1,4824 \end{bmatrix}^{-1} \begin{bmatrix} 0,1302 & 0,1302 \\ 2,5168 & -2,5168 \end{bmatrix}$$

$$T = \begin{bmatrix} 0,1179 & 0,1179 \\ 1,6978 & -1,6978 \end{bmatrix}$$

Multiplicando a matriz B por T^{-1} obtém-se a nova matriz de entrada (B') para o sistema, dada por: $BT^{-1} = B(-B^{-1}A) = -A$. Assim, é possível escrever o modelo dinâmico dos atuadores do robô com entradas e_v e e_ω :

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -1,1046 & 0,0000 \\ 0,0000 & -1,4824 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} + \begin{bmatrix} 1,1046 & 0,0000 \\ 0,0000 & 1,4824 \end{bmatrix} \begin{bmatrix} e_v \\ e_\omega \end{bmatrix}$$

Assim o modelo dinâmico para o robô, seguindo a representação dada em 5.1, pode ser escrito como:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \\ \dot{s} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -1,1046 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & -1,4824 & 0,0000 & 0,0000 \\ 1,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 1,0000 & 0,0000 & 0,0000 \end{bmatrix} \begin{bmatrix} v \\ \omega \\ s \\ \theta \end{bmatrix} + \begin{bmatrix} 1,1046 & 0,0000 \\ 0,0000 & 1,4824 \\ 0,0000 & 0,0000 \\ 0,0000 & 0,0000 \end{bmatrix} \begin{bmatrix} e_v \\ e_\omega \end{bmatrix} \quad (5.4)$$

A controlabilidade do sistema é verificada através do posto da matriz de controlabilidade dada por:

$$\mathcal{C} = [\mathbf{B} \ \mathbf{AB} \ \dots \ \mathbf{A}^{n-1}\mathbf{B}]$$

Assim, para este sistema, tem-se que o sistema é controlável pois o $\text{posto}(\mathcal{C}) = 4$. No entanto, este sistema apresenta dois pólos na origem, uma referente ao modelo de deslocamento linear e outro referente ao modelo de deslocamento angular, que são observáveis, porém apenas por uma das saídas do sistema s ou θ , devido ao desacoplamento considerado no modelo.

A partir da representação em espaço de estado é possível gerar uma representação por matriz de funções transferência, da forma:

$$\mathbf{y} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \mathbf{u} \quad (5.5)$$

onde \mathbf{y} representa o vetor de saídas do sistema e \mathbf{u} representa o vetor de entradas. Para o caso do robô simétrico, conforme modelo levantado, e considerando as saídas do sistema como “ s ” e “ θ ”, obter-se-á, após a transformação para função de transferência, um modelo na forma:

$$\begin{bmatrix} s \\ \theta \end{bmatrix} = \begin{bmatrix} G_s & 0 \\ 0 & G_\theta \end{bmatrix} \begin{bmatrix} e_v \\ e_\omega \end{bmatrix}$$

Assim, transformando o modelo da representação dada por (5.4) para a representação dada por (5.5), obtém-se:

$$\begin{bmatrix} s \\ \theta \end{bmatrix} = \begin{bmatrix} \frac{1,1046}{s^2+1,1046s} & 0 \\ 0 & \frac{1,4824}{s^2+1,4824s} \end{bmatrix} \begin{bmatrix} e_v \\ e_\omega \end{bmatrix} \quad (5.6)$$

Com isto, tem-se diretamente as funções de transferência:

$$G_s(s) = \frac{1,1046}{s^2 + 1,1046s} \quad (5.7)$$

e

$$G_\theta(s) = \frac{1,4824}{s^2 + 1,4824s} \quad (5.8)$$

As funções de transferência de $G_s(s)$ e $G_\theta(s)$ são de segunda ordem pois foi considerado o desacoplamento perfeito entre os movimentos angular e linear. Conforme comentado anteriormente, nos casos em que o grau de acoplamento é baixo, pode-se, de maneira simplificada, projetar controladores monovariáveis para atuar sobre os movimentos de translação (ao longo de s) e rotação (ao longo de θ), separadamente (ver figura 5.3). Com esta premissa, pode-se escolher o tipo de controlador que atenda aos requisitos dinâmicos exigidos pela tarefa a ser executada pelo robô.

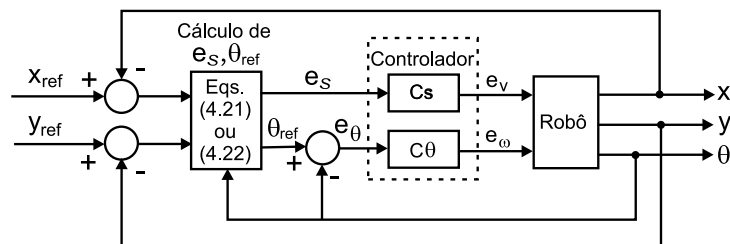


Figura 5.1: Diagrama de blocos do sistema de controle de posição utilizando controladores monovariáveis.

5.2 Projeto dos Controladores

A partir das funções de transferência G_s e G_θ verificou-se que, para os parâmetros utilizados do robô, essas funções, em malha fechada com realimentação unitária, são BIBO (*Bounded Input Bounded Output* - Entrada Limitada Saída Limitada) estáveis. Desta forma, são apresentadas a seguir simulações (em MATLAB) para cada uma das funções de transferência (resposta ao degrau unitário em malha fechada), assim como os parâmetros de tempo de subida e sobre-sinal.

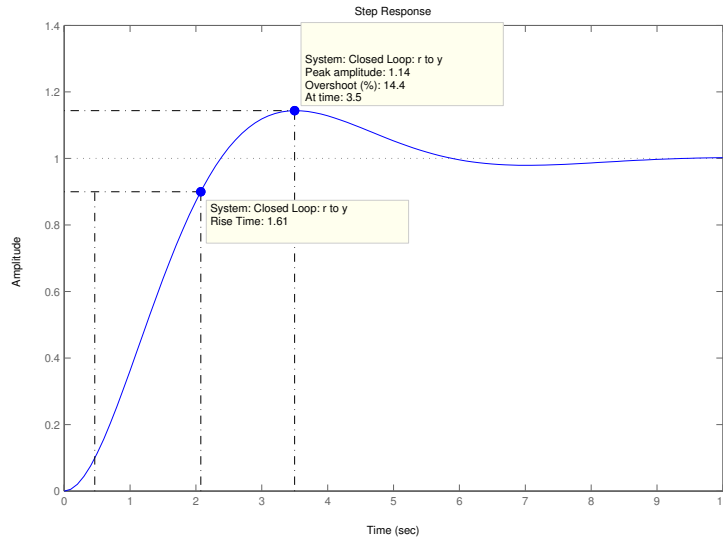


Figura 5.2: Resposta ao degrau da função de transferência G_s em malha fechada.

A resposta ao degrau unitário para G_s em malha fechada apresenta um tempo de subida de 1,61 segundos e um sobre-sinal de 14,4%.

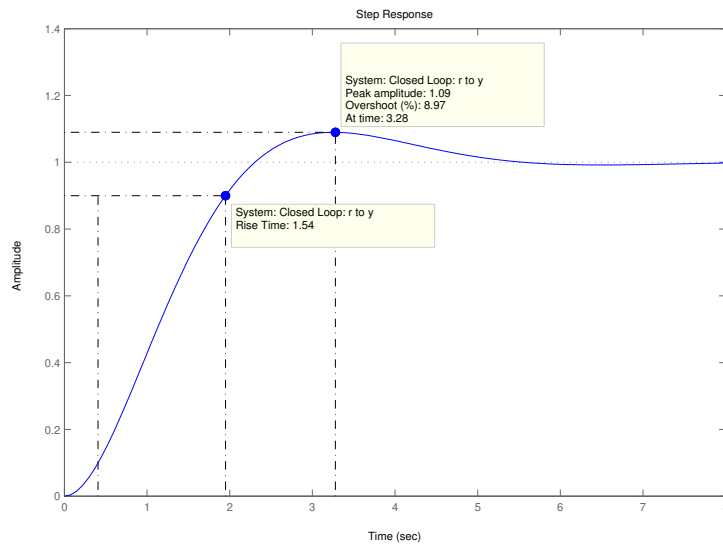


Figura 5.3: Resposta ao degrau da função de transferência G_θ em malha fechada.

A resposta ao degrau unitário para G_θ em malha fechada apresenta um tempo de subida de

1,54 segundos e um sobre-sinal de 8,97%.

Como as funções de transferência G_s e G_θ são BIBO estáveis, o projeto de um controlador é necessário apenas para permitir ajustes na dinâmica desejada (características como tempo de subida e sobre-sinal) para o sistema, conforme será abordado nas seções que seguem.

5.2.1 Controle de Posição

Antes de prosseguir com o projeto dos controladores, é interessante lembrar que, no caso de controle de posição, considera-se uma referência fixa (x_d e y_d) no espaço cartesiano. Com isto, tem-se que a garantia de convergência dos erros

$$\begin{aligned} e_s &= \Delta l \cdot \cos(\Delta\phi) \\ e_\theta &= \Delta\phi \end{aligned} \quad (5.9)$$

para zero implica na garantia de alcance do objetivo de controle, conforme apresentado no capítulo 4. Assim, para o caso em que a dinâmica do sistema em malha fechada (com realimentação unitária) é *BIBO* estável, o sistema convergirá para o objetivo de controle quando forem realimentados os sinais referentes à configuração do robô (x, y e θ), gerando os erros definidos em (5.9) – é necessário que as funções de transferência dos movimentos linear e angular apresentem pólos na origem para garantir a convergência dos erros para zero. A figura 5.4 a seguir apresenta o resposta do sistema robótico modelado utilizando a estratégia de controle de posição, sem a introdução de controladores, para a configuração inicial $x = y = \theta = 0$ e configuração desejada $x_{ref} = y_{ref} = 1$. Assim, quando a dinâmica do robô em malha fechada é *BIBO* estável e

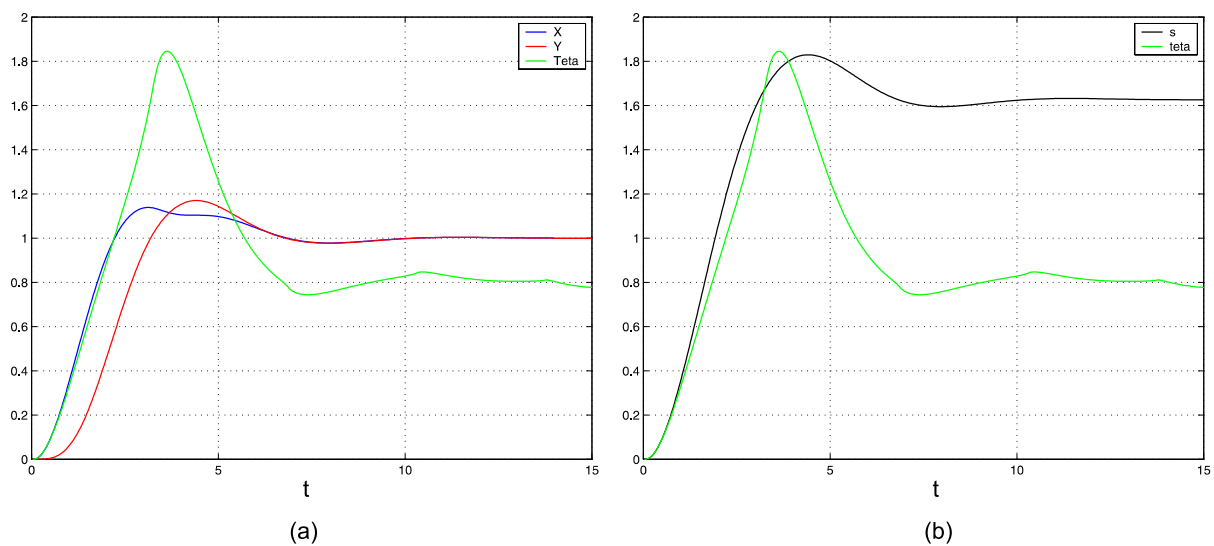


Figura 5.4: (a) Resposta em x, y e θ do sistema robótico com realimentação de erros, sem controlador. (b) Resposta em s e θ do sistema robótico.

quando funções de transferência dos movimentos linear e angular apresentarem pólos na origem, independentemente das respostas dinâmicas de s e θ , haverá convergência dos erros para zero e conseqüentemente o alcance do objetivo de controle.

Apesar do exposto, quanto melhor for a dinâmica do movimento angular (rápido tempo de subida com baixo sobre-sinal) em relação à dinâmica do movimento linear, melhor será a resposta do sistema, visto que o robô tenderá a corrigir mais rapidamente sua orientação em relação à posição desejada (x_d, y_d) e prosseguir em movimento linear até o objetivo. Desta forma, é interessante projetar controladores que permitam baixo sobre-sinal tanto para o movimento angular quanto linear, e uma resposta mais rápida do movimento angular em relação ao linear.

Conforme já discutido qualquer tipo de controlador que garanta a convergência dos erros para zero pode ser utilizado em conjunto com a estratégia proposta. Neste caso específico, onde a dinâmica de malha fechada é *BIBO* estável, até mesmo controladores proporcionais (P) puros podem ser utilizados.

Devido ao desacoplamento do sistema apresentado anteriormente, é possível utilizar técnicas de projeto de controladores *SISO* para os movimentos angular e linear separadamente. A técnica de projeto utilizada foi o lugar das raízes (*root-locus*) com observação da resposta da planta em malha fechada para entrada degrau unitário. Foi utilizada a ferramenta *rltool* do MATLAB para o projeto. Decidiu-se implementar um controlador PID para o movimento angular de modo a evitar o erro em regime, devido a não-linearidades não modeladas do sistema (mesmo com a existência de uma ação integral presente no modelo), e um controlador PD para o movimento linear, pois obteve-se um menor sobre-sinal com controlador PD para o movimento linear.

Controlador Angular

Considerando os pontos levantados anteriormente, segue o projeto para os controladores angular e linear. Primeiramente verificou-se o lugar das raízes para a função de transferência referente ao movimento angular do robô, dada por

$$G_{\theta}(s) = \frac{1,4824}{s(s + 1,14824)} \quad (5.10)$$

O lugar das raízes para esta função de transferência é apresentado na figura 5.5. A partir deste foi feito o posicionamento de um pólo na origem e dois zeros reais de forma a atingir a resposta desejada para a planta. O lugar das raízes após o posicionamento é dado na figura 5.6.

A forma geral de um controlador PID é dada por:

$$C(s) = K_p \left(\tau_d s + \frac{1}{\tau_i s} + 1 \right)$$

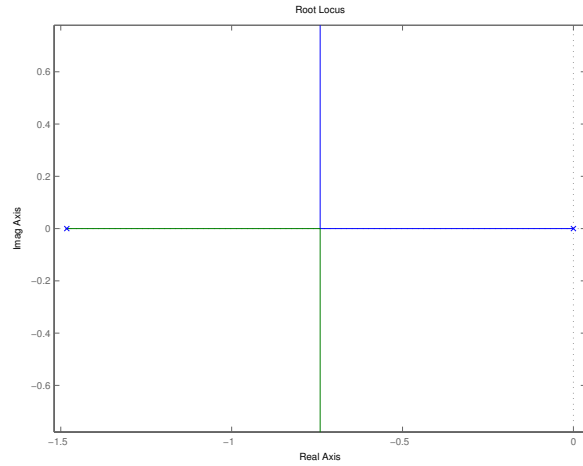


Figura 5.5: Lugar das raízes para $G_\theta(s)$.

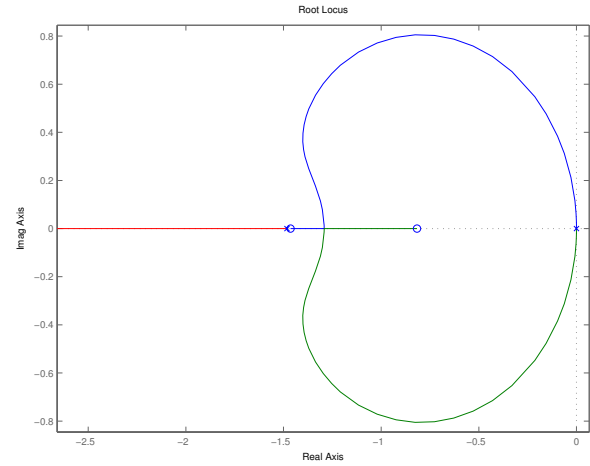


Figura 5.6: Lugar das raízes para $G_\theta(s)$ com controlador $C(s)$ tipo PID.

O controlador PID obtido para o movimento angular é apresentado na equação 5.13 a seguir.

$$C_\theta(s) = 5,9145 \left(0,4385s + \frac{1}{1,9079s} + 1 \right) \quad (5.11)$$

A partir da implementação do controlador PID projetado foram simuladas as condições com e sem controlador, apresentadas na figura 5.7. É possível perceber a melhora obtida no tempo de subida (de 1,54 segundos para 0,396 segundos), no entanto ao custo de um aumento no sobre-sinal (de 8,97% para 12,0%).

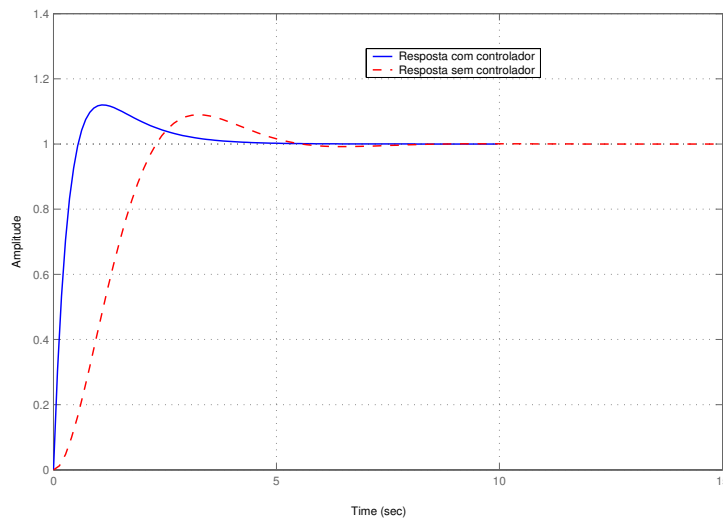


Figura 5.7: Resposta do sistema (movimento angular) em malha fechada, com e sem o controlador PID projetado.

Controlador Linear

Primeiramente verificou-se o lugar das raízes para a função de transferência referente ao movimento angular do robô, dada por:

$$G_s(s) = \frac{1,1046}{s(s + 1,1046)} \quad (5.12)$$

O lugar das raízes para esta função de transferência é apresentado na figura 5.8. Para esta função de transferência optou-se por um controlador PD. Assim, foi feito o posicionamento de um zero real, juntamente com o ajuste de ganho de forma a atingir a resposta desejada para a planta. O lugar das raízes após o posicionamento é dado na figura 5.9.

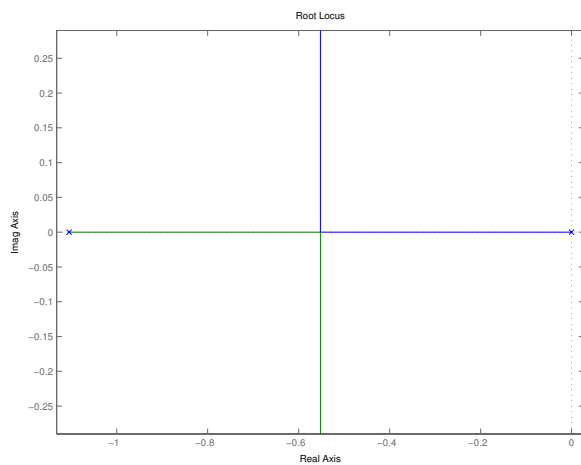


Figura 5.8: Lugar das raízes para $G_s(s)$.

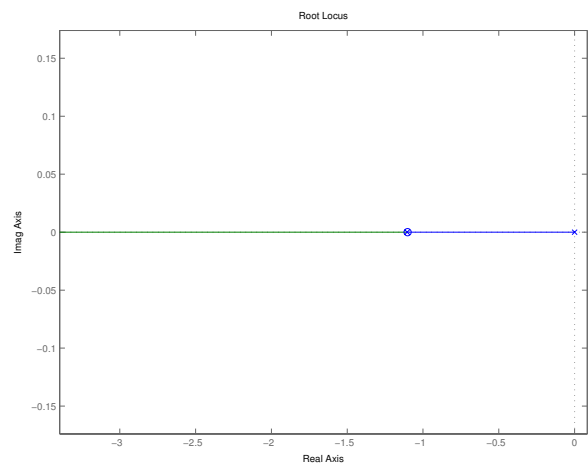


Figura 5.9: Lugar das raízes para $G_s(s)$ com controlador $C(s)$ tipo PD.

A forma geral para o controlador PD pode ser obtida da forma geral do controlador PID suprimindo-se o termo integral. A função de transferência para o controlador linear obtido é dada por:

$$C_s(s) = 0,8(0,9074s + 1) \quad (5.13)$$

A partir da implementação do controlador PD projetado foram simuladas as condições com e sem controlador, apresentadas na figura 5.10. O objetivo principal era reduzir o sobre-sinal sem perda no tempo de subida. Pode-se perceber que houve a redução do sobre sinal a 0%, no entanto houve um pequeno aumento no tempo de subida (compensado por um tempo de estabilização menor).

Resultados simulados e experimentais serão apresentados no capítulo 6 abordando exemplos com diversas configurações iniciais e finais.

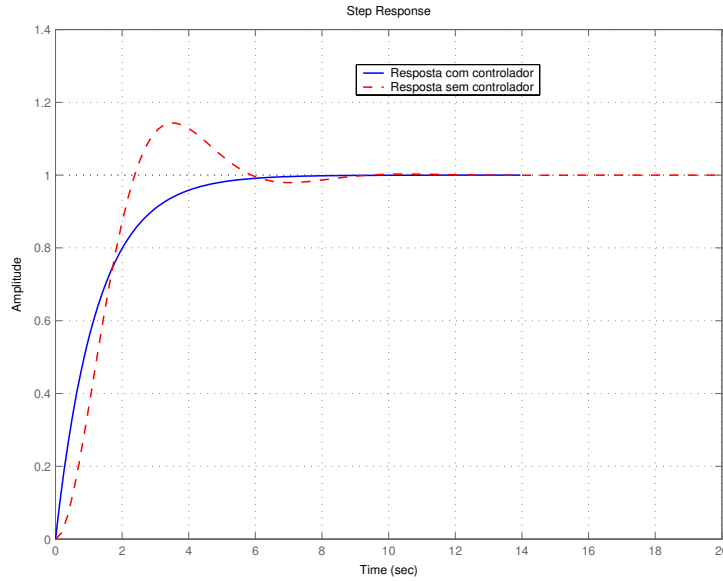


Figura 5.10: Resposta do sistema (movimento linear) em malha fechada, com e sem o controlador PD projetado.

5.2.2 Controle de Orientação

Diferentemente do problema de controle de posição, o problema de controle de orientação envolve a utilização de uma referência móvel, e isto tem impacto direto no aspecto de estabilidade e convergência da estratégia proposta para este problema. Não procedeu-se uma análise mais aprofundada sobre o sistema não-linear abordado juntamente com a estratégia de controle proposta, no entanto, algumas observações foram realizadas durante experimentos e que são válidas para a implementação do controlador de orientação.

O sistema de geração de referência móvel pode levar ao fenômeno de auto-oscilação em alguns casos. Isto pode ocorrer, quando os controladores levarem a uma estabilização mais rápida do sub-sistema linear em relação ao sub-sistema angular. A estratégia de controle de orientação é baseada em uma estratégia de controle de posição onde o erro angular não deve, necessariamente, convergir para zero (vide capítulo 4), no entanto, para que a estratégia de controle de orientação funcione, não somente é necessário que o erro angular convirja para zero, mas também que isto ocorra em tempo inferior ao tempo de estabilização do controlador linear. De uma forma mais amigável pode-se dizer que a referência móvel deve servir apenas para corrigir o ângulo (como um volante) do robô ao longo da trajetória até o objetivo desejado no espaço cartesiano. Assim, se o tempo de estabilização do movimento linear é inferior ao tempo de estabilização do movimento angular, o objetivo passa a ser atingir a referência móvel o mais rápido possível, podendo gerar oscilações. Por isso, deve haver um compromisso entre essas dinâmicas (linear e angular) de forma a garantir que o tempo de estabilização do movimento angular sempre seja

inferior ao tempo de estabilização linear. As considerações expostas são baseadas em observações obtidas através de várias simulações com diversas combinações de controladores e requerem um estudo mais aprofundado para determinar não apenas qualitativamente, mas quantitativamente, os limites relacionados à dinâmica do sistema. As figuras 5.11 e 5.12 mostram o fenômeno de auto-oscilação que pode ocorrer de acordo com as dinâmicas estabelecidas para o sistema. No caso apresentado foram utilizadas as mesmas configurações iniciais e finais, porém, no caso em não ocorre oscilação, adotou-se uma dinâmica mais lenta para o movimento linear. Observando

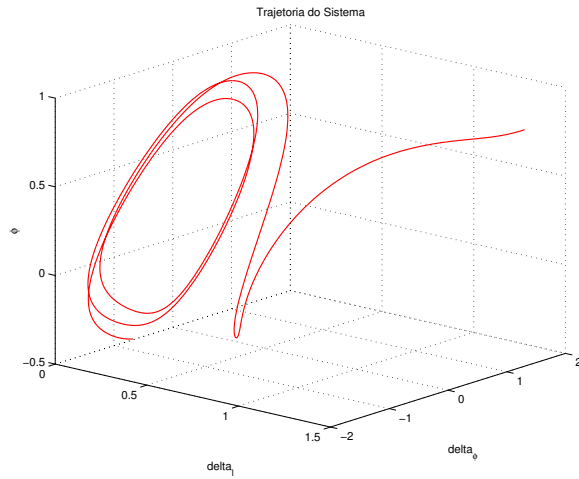


Figura 5.11: Evolução das variáveis de estado Δl , $\Delta\phi$ e ϕ para condição de oscilação.

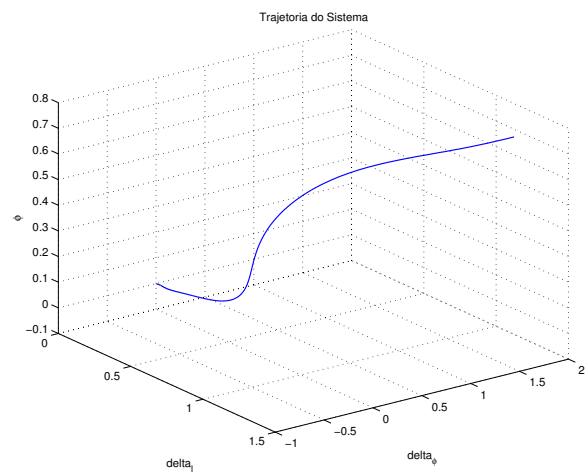


Figura 5.12: Evolução das variáveis de estado Δl , $\Delta\phi$ e ϕ para condição sem oscilação.

a figura 5.12 pode-se perceber a convergência do sistema para $\Delta l = \Delta\phi = \phi = 0$, diferentemente do resultado apresentado na figura 5.11. As figuras 5.13 e 5.14 mostram o que ocorre com as variáveis de estado x , y e θ em ambos os casos.

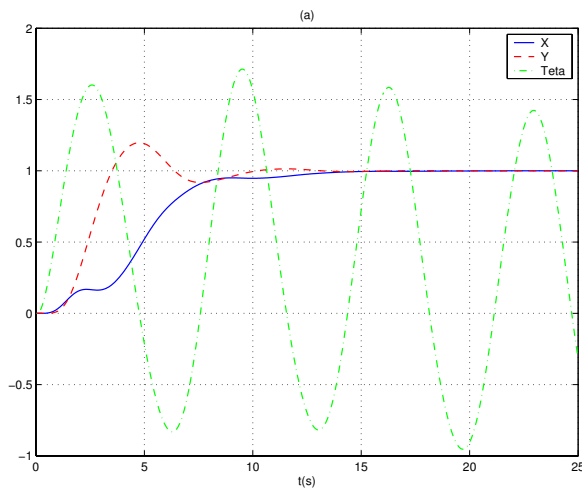


Figura 5.13: Evolução das variáveis de estado x , y e θ para condição de oscilação.

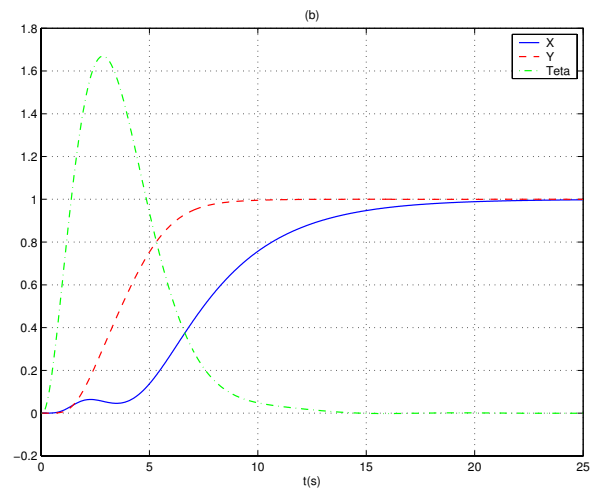


Figura 5.14: Evolução das variáveis de estado x , y e θ para condição sem oscilação.

Apesar dos aspectos abordados com relação à estratégia de controle de orientação proposta, deve-se lembrar que, se for utilizado um gerador de trajetórias acoplado ao controlador de posição, este problema pode ser eliminado. Conforme já discutido, um gerador de trajetórias gera uma trajetória para o robô baseando-se nos aspectos cinemático (restrições não-holonômicas) e dinâmicos do sistema.

Assim, baseado nos comentários em relação ao controle de orientação pode-se utilizar os mesmos controladores projetados para o controle de posição, visto que estes se encaixam nos requisitos levantados empiricamente. A figura 5.15 apresenta o diagrama de blocos para o controle de orientação.

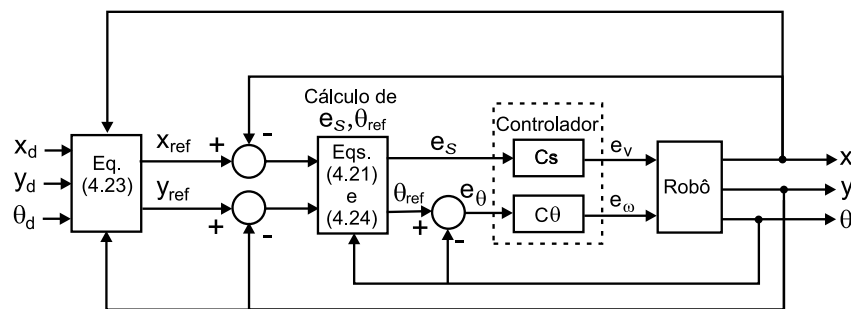


Figura 5.15: Diagrama de blocos do sistema de controle de orientação utilizando controladores monovariáveis.

5.3 Conclusão

Conforme discutido previamente o projeto dos controladores dinâmicos requer o conhecimento do sistema. Desta forma, sempre é necessário ajustar os controladores de acordo com o sistema robótico a ser operado. No entanto, é possível acoplar a estratégia proposta a um identificador *online* do sistema e adotar técnicas de controle que permitam absorver variações paramétricas do modelo. Este trabalho deverá evoluir para incorporação destas técnicas.

Conforme apresentado neste capítulo os controladores dinâmicos para as estratégias de controle de posição e orientação do robô móvel foram projetados utilizando a técnica clássica de projeto através do lugar das raízes. Para o projeto, foram obtidas primeiramente as funções de transferência relativas aos movimentos linear e angular. Desta forma, após a obtenção das funções de transferência desacopladas, foram projetados controladores PID (Proporcional–Integral–Derivativo) para a malha de controle de orientação e PD (Proporcional–Derivativo) para o controle de translação. Estes foram projetados com base em algumas características desejadas, que dizem respeito aos parâmetros de desempenho (sobre-sinal, tempo de subida, tempo de estabilização) em malha fechada do sistema. Após o desenvolvimento dos controlado-

res contínuos obtidos, foi feita sua implementação discreta no algoritmo de controle (utilizando aproximação de Euler).

Deve-se notar que, apesar do sistema apresentar dois controladores distintos, estes operam simultaneamente. Desta forma, o robô executa trajetórias suaves, combinando os movimentos lineares e angulares, conforme será apresentado no capítulo 6.

Deve-se lembrar que apesar da escolha feita para os controladores, outros tipos podem ser utilizados e, em alguns casos, isto é necessário (por exemplo, quando houver um grau de acoplamento elevado entre as dinâmicas do movimento linear e angular).

Capítulo 6

Resultados

Neste capítulo são apresentados e discutidos resultados obtidos através de simulação e experimentos para as estratégias de controle de posição e orientação propostas. As simulações e os experimentos abordam diversas configurações finais para o robô e reforçam aspectos apontados durante o desenvolvimento das estratégias. As simulações foram realizadas através do simulador implementado em MATLAB. Neste simulador foram implementados os controladores angular e linear apresentados no capítulo 5. Os resultados experimentais foram obtidos através da plataforma descrita no capítulo 3.

6.1 Controle de Posição

A seguir são apresentados os resultados simulados e experimentais para o controle de posição do robô. Também serão abordados casos de seguimento de trajetória utilizando o controle de posição. Em cada figura de saída são apresentadas as evoluções das seguintes variáveis: sinal de controle, trajetória do robô no espaço cartesiano, erros linear e angular, e variáveis de configuração do robô (x , y e θ). Os seguintes casos são abordados:

- Simulação do controle de posição, sem a modificação da estratégia de controle proposta na seção 4.2.2 do capítulo 4.
- Simulação do controle de posição, com a modificação da estratégia de controle proposta na seção 4.2.2 do capítulo 4.
- Simulação do controle de posição aplicado ao seguimento de caminho.
- Simulação do controle de posição incluindo todas fontes de ruído, incertezas e atraso de tempo disponibilizadas pelo simulador.

- Resultados experimentais da estratégia de controle de posição e sua aplicação ao seguimento de trajetória.

Para as simulações envolvendo o controle de posição será considerada a configuração inicial $q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$. Os resultados apresentados serão discutidos no decorrer desta seção.

6.1.1 Controle de Posição com Formulação Original

A seguir é apresentada a primeira simulação considerando a formulação original da estratégia de controle de posição. Como no caso de controle de posição a orientação final não é controlada, esta será suprimida do vetor de configuração final para os casos simulados.

- Configuração final $q = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$

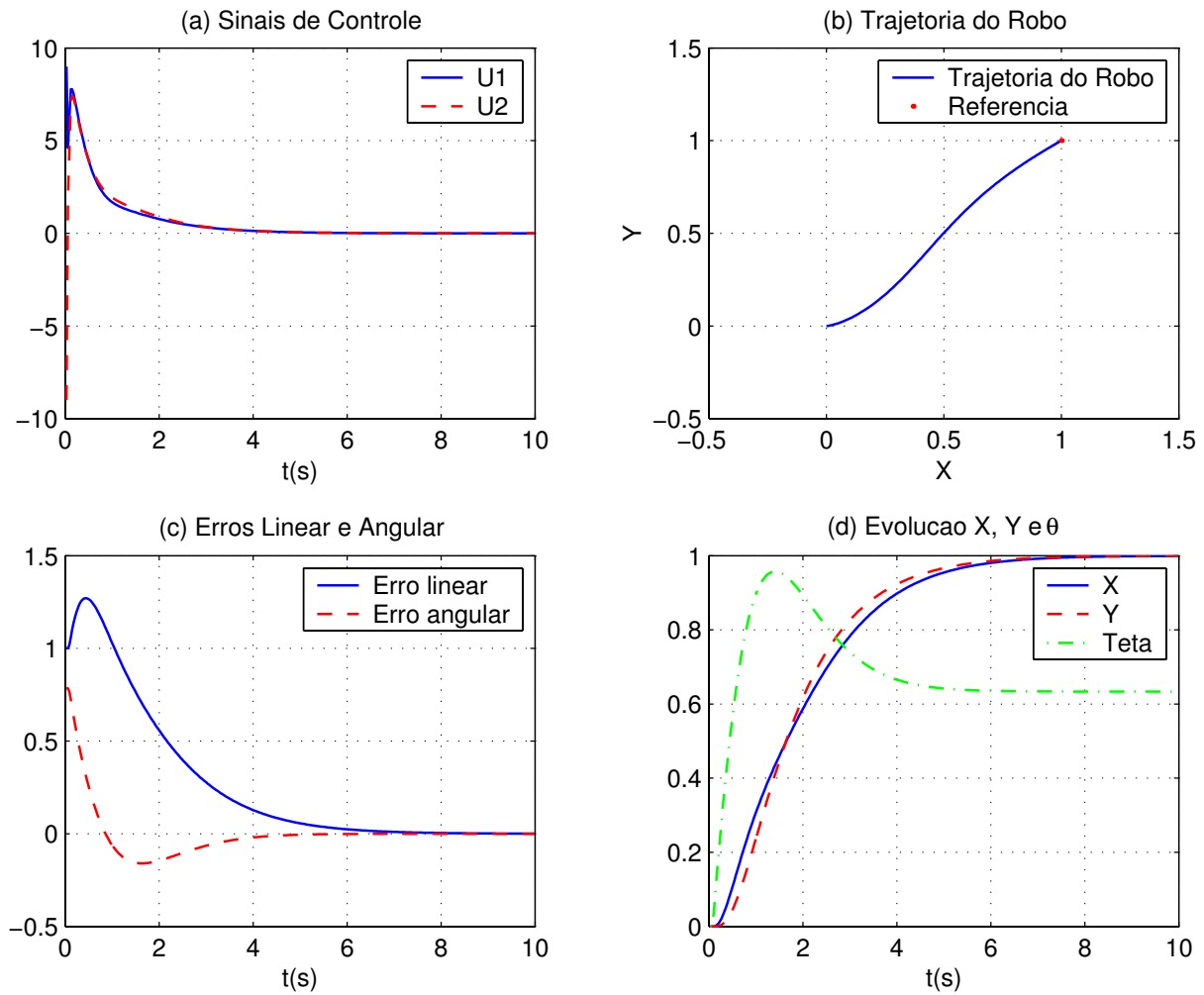


Figura 6.1: Simulação do controle de posição - caso 1.

Observando a figura 6.1 pode-se perceber a convergência dos erros para zero assim como a evolução da trajetória do robô para a posição desejada, conforme esperado. Nesta simulação não foi possível identificar diferenças com relação à modificação proposta para a estratégia de controle de posição, visto que o controlador linear foi projetado de forma a eliminar o sobre-sinal. Assim, posteriormente será comparado o desempenho do sistema quando o sobre-sinal está presente.

- Configuração final $q = \begin{bmatrix} -1 & -1 \end{bmatrix}^T$

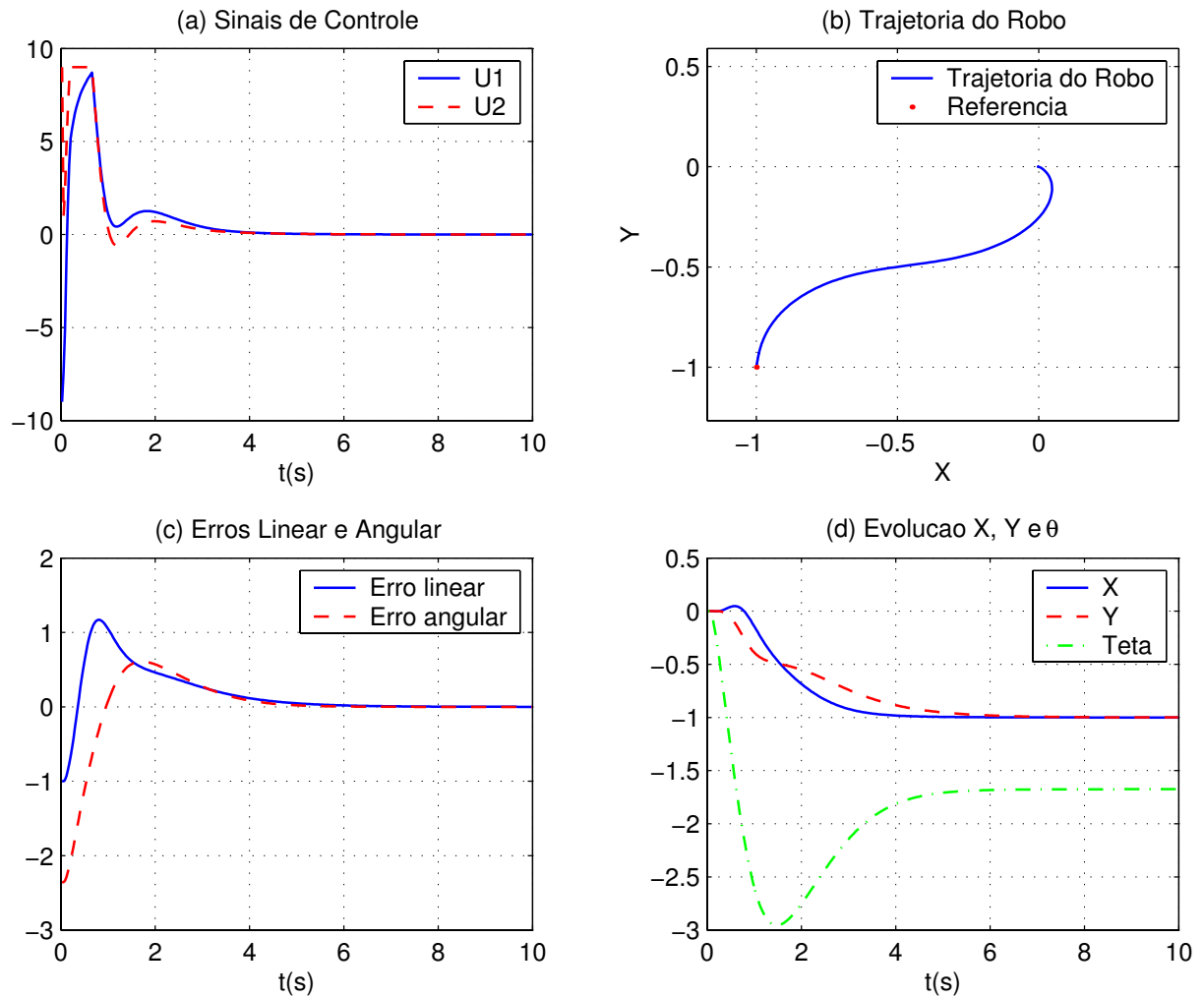


Figura 6.2: Simulação do controle de posição - caso 2.

A figura 6.2 mostra que o robô tenta atingir o objetivo de forma orientada para este, isto é, “de frente”. Em alguns casos isto pode ser desejado porém, dependendo da configuração inicial, pode ser necessário um tempo maior para estabilização em torno do objetivo.

6.1.2 Controle de Posição com Modificação

A seguir é apresentada a simulação com a mesma configuração final utilizada em 6.2. O objetivo é mostrar a diferença do movimento quando é empregada a modificação na estratégia de controle (ver item 4.2.2 do capítulo 4).

- Configuração final $q = \begin{bmatrix} -1 & -1 \end{bmatrix}^T$

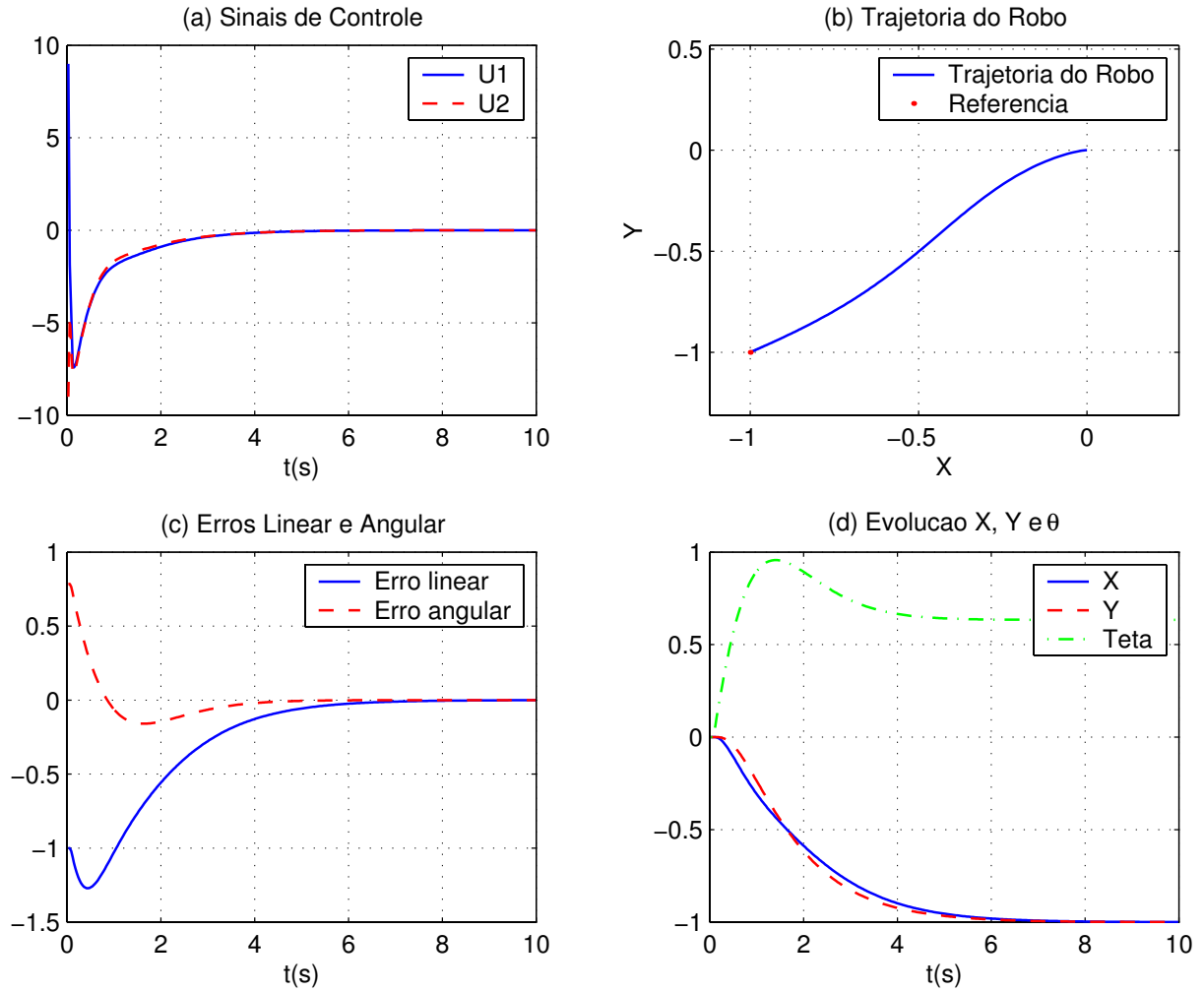


Figura 6.3: Simulação de controle de posição com modificação proposta.

Pode-se perceber que, da mesma forma como na implementação da formulação original, os erros linear e angular convergem para zero garantindo o alcance do objetivo de controle. Pode-se também perceber, que em ambos os casos o sinal de controle se apresenta de forma suave, isto é, sem chaveamentos. Na seção 6.1.4 serão apresentados resultados incluindo fontes de ruído e incertezas ao sistema de controle.

A figura 6.4 apresenta uma comparação da trajetória do robô para os casos com e sem a modificação proposta (para uma mesma configuração final) e utilizando um controlador linear

que apresenta sobre-sinal. A presença do sobre sinal permite a visualização do efeito provocado pela modificação na estratégia de controle de posição. Desta forma, é possível verificar que no caso em que a modificação é implementada há uma aproximação mais “inteligente” do objetivo. Apesar da modificação proposta para correção do erro angular trazer um chaveamento para a

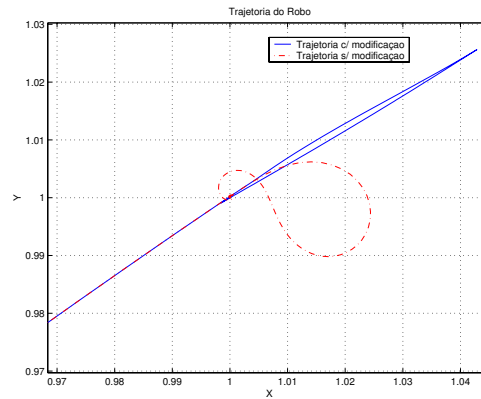


Figura 6.4: Trajetórias para uma mesma configuração desejada, com e sem a modificação na estratégia de controle.

definição do erro angular, este não gera um impacto acentuado nos sinais de controle, visto que o chaveamento não é constante, isto é, a lei de controle não é baseada apenas no chaveamento para estabilização do sistema. Além disso, dependendo do projeto dos controladores, é possível eliminar o efeito de chaveamento através da eliminação do sobre-sinal na dinâmica do movimento linear.

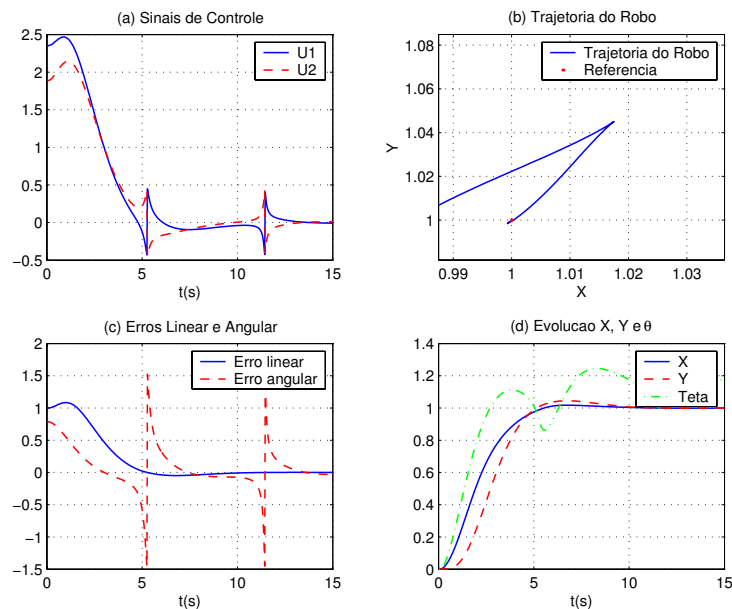


Figura 6.5: Simulação demonstrando chaveamento do sinal de controle.

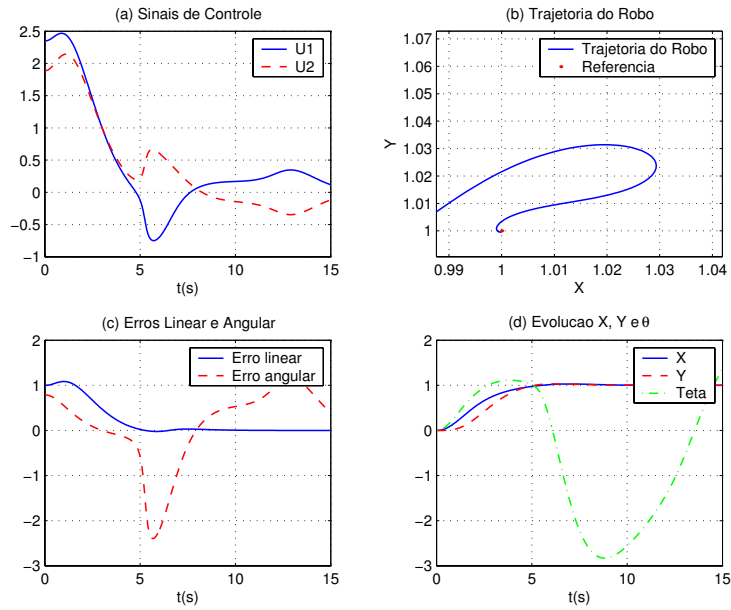


Figura 6.6: Simulação demonstrando sinal de controle suave.

As simulações apresentadas nas figuras 6.5 e 6.6 mostram a diferença dos sinais de controle para cada uma das estratégias de controle de posição.

6.1.3 Controle de Posição para Seguimento de Trajetória

Conforme discutido previamente, o controlador de posição pode ser implementado juntamente com um gerador de trajetórias. Na simulação que segue foi implementada uma referência móvel gerando uma trajetória circular, para permitir a verificação deste tipo de implementação. O resultado apresentado na figura 6.7 confirma a condição de implementação proposta. No

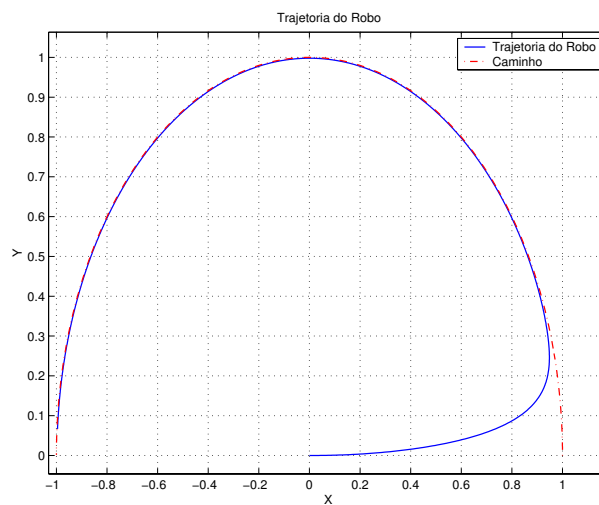


Figura 6.7: Trajetória circular gerada e evolução do robô utilizando a estratégia de controle de posição.

entanto, deve-se observar que a trajetória gerada deve respeitar as restrições não-holonômicas

do robô.

Observando a trajetória do robô é possível confirmar que, apesar de não haver a designação sobre o ângulo a ser seguido ao percorrer a trajetória, este é automaticamente corrigido, visto que o robô, ao seguir a referência imposta pela trajetória, necessariamente realiza as correções angulares.

6.1.4 Controle de Posição com Ruídos e Incertezas

O simulador implementado permite incorporar aos resultados algumas fontes de ruído e incertezas que podem interferir no desempenho do controlador (considerando a plataforma na qual é proposta a aplicação), a saber: ruídos de processo, ruídos de medição, ruídos introduzidos pelo efeito de quantização da posição do robô e orientação (derivada a partir da posição), visto que são utilizadas imagens digitalizadas para localização, efeitos de quantização do sinal de controle, visto que há apenas 128 níveis para o sinal de controle, e, finalmente, o atraso puro de tempo, presente devido ao cálculo *off-board* do sinal de controle e sua transmissão via rádio para o veículo. O objetivo neste caso é observar o comportamento do sistema frente a problemas que são observados em sua implementação prática.

Os resultados apresentados na figura 6.8 mostram que as fontes de ruído e incerteza têm forte impacto principalmente no sinal de controle. Percebe-se que quanto mais próximo o robô se aproxima do objetivo, pior fica o sinal de controle. Isto ocorre pois, neste caso, a referência angular $\theta_{ref} = \phi$ está sendo calculada a partir de valores quantizados de posição e orientação do robô e com a presença de ruídos. Desta forma quanto mais próximo do objetivo maiores ficam as incertezas relativas ao posicionamento do robô e ao ângulo de orientação em relação ao objetivo (ϕ).

Portanto, pode-se observar que, na prática, estes efeitos podem prejudicar o desempenho do controlador e que a implementação de filtros para os sinais obtidos do sistema de localização pode ser necessária, principalmente quando tratando do controle de orientação.

Para atenuar o efeito de ruídos no sinal de controle foram introduzidos filtros, comumente implementados quando o controlador envolve ganho derivativo. Assim, mesmo na presença dos ruídos citados consegue-se atingir um bom desempenho do sistema.

A partir das figuras 6.9 e 6.10 pode-se ter uma comparação melhor das distorções provocadas no sinal de controle quando são considerados os efeitos de quantização e ruídos.

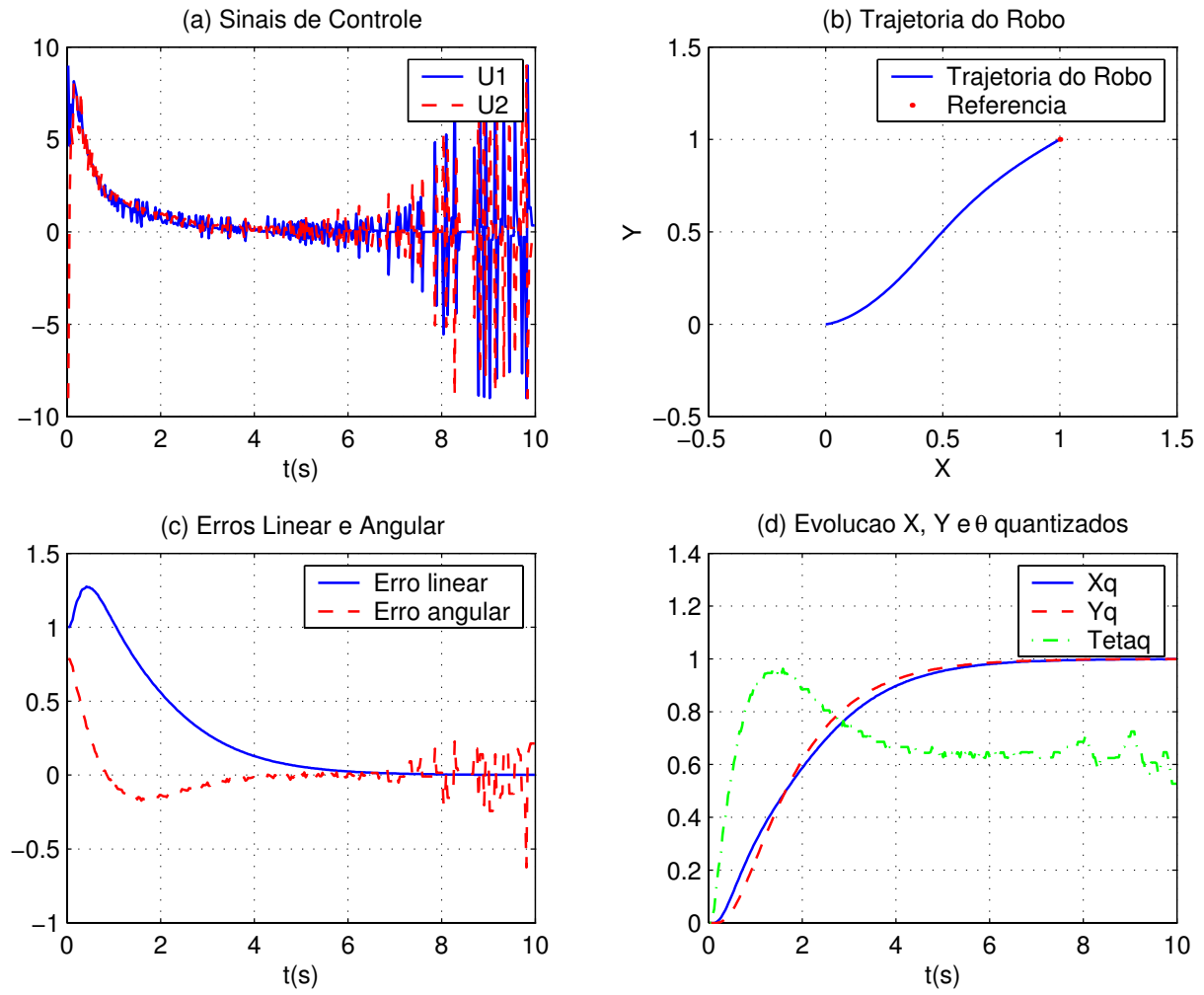


Figura 6.8: Controle de posição com introdução de fontes de ruído e incertezas.

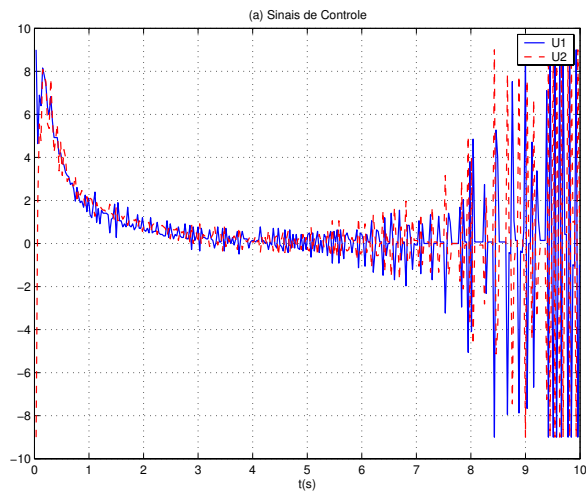


Figura 6.9: Sinal de controle na presença de ruídos e incertezas.

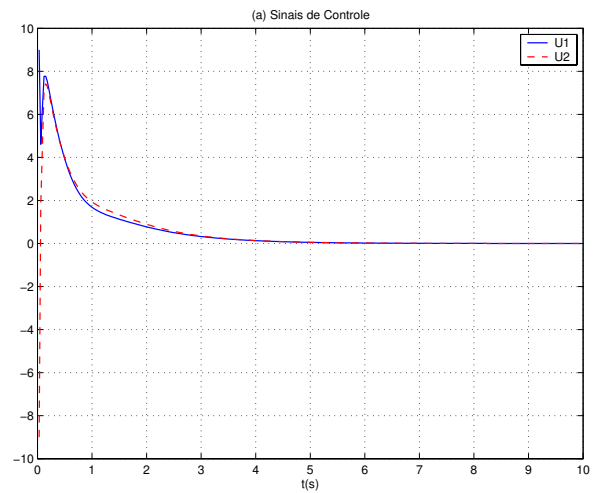


Figura 6.10: Sinal de controle na ausência de ruídos e incertezas.

6.1.5 Resultados Experimentais para a Estratégia de Controle de Posição

Os resultados experimentais para controle de posição foram obtidos através da plataforma descrita no capítulo 3 e demonstram o desempenho da estratégia de controle posição. Primeiramente são apresentados resultados experimentais referentes à estratégia de controle de posição. Conforme apresentado nos gráficos da figura 6.11, foram feitos ensaios nos quais um conjunto de referências periódicas no espaço cartesiano é passado ao sistema de controle.

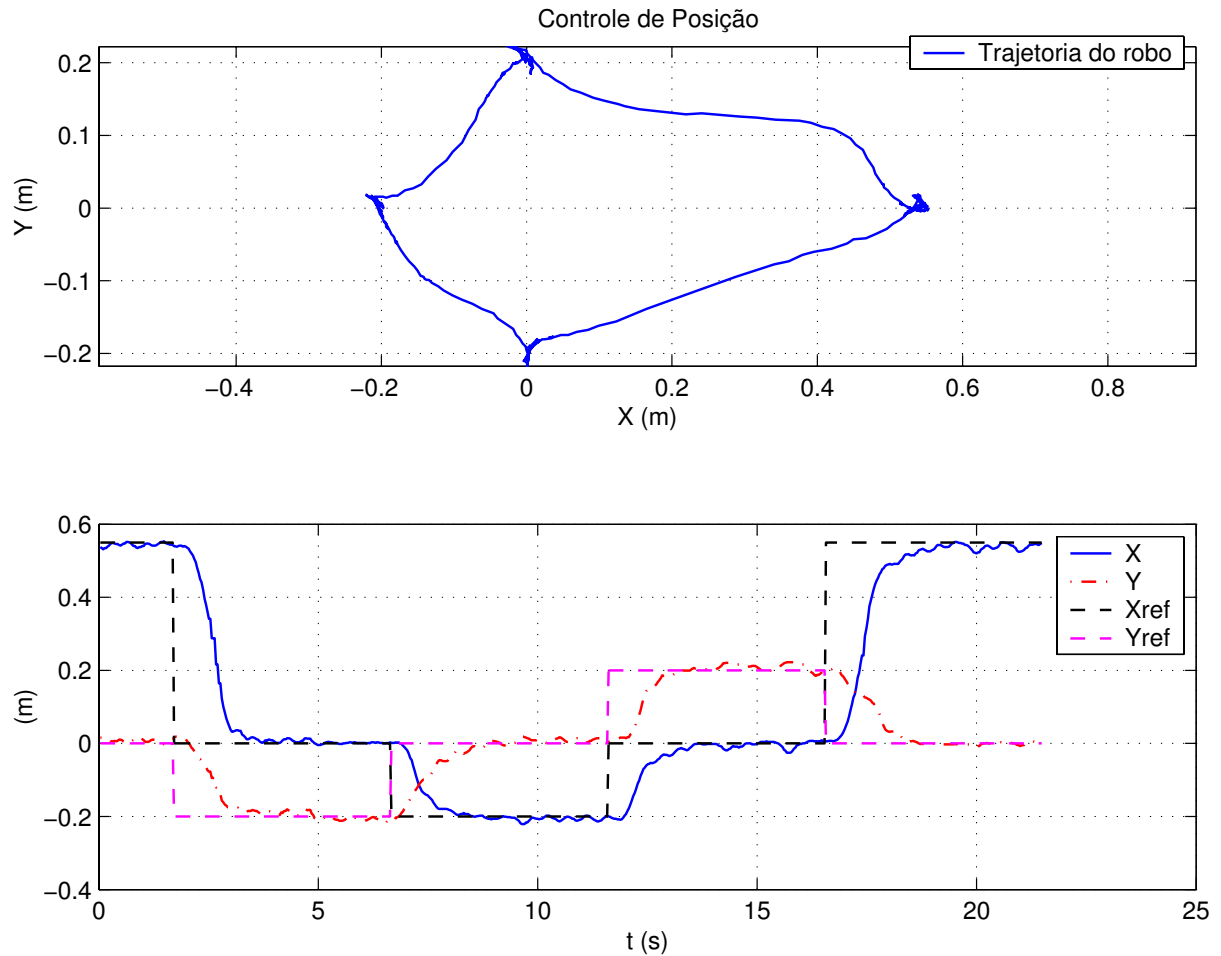


Figura 6.11: Resultados experimentais para a estratégia de controle de posição.

Na figura 6.11 são apresentados dois gráficos. O superior indica a trajetória percorrida pelo robô para o alcance dos objetivos de controle e o gráfico inferior apresenta a evolução das variáveis de controle (x e y) no tempo, assim como suas respectivas referências.

Em seguida foram realizados ensaios para demonstrar a capacidade da utilização da estratégia de controle acoplada a um gerador de caminho ou trajetória. A figura 6.12 apresenta os resultados de um ensaio para seguimento de uma trajetória circular (com raio de 20 cm) com período inicial de 4π segundos onde a partir de um determinado tempo o período é diminuído para

2π segundos. Deve-se lembrar que, pelo fato de estar sendo utilizado apenas a estratégia de controle de posição, a orientação do robô é determinada pelo perfil apresentado pela trajetória obedecendo as restrições não-holonômicas de movimento do robô.

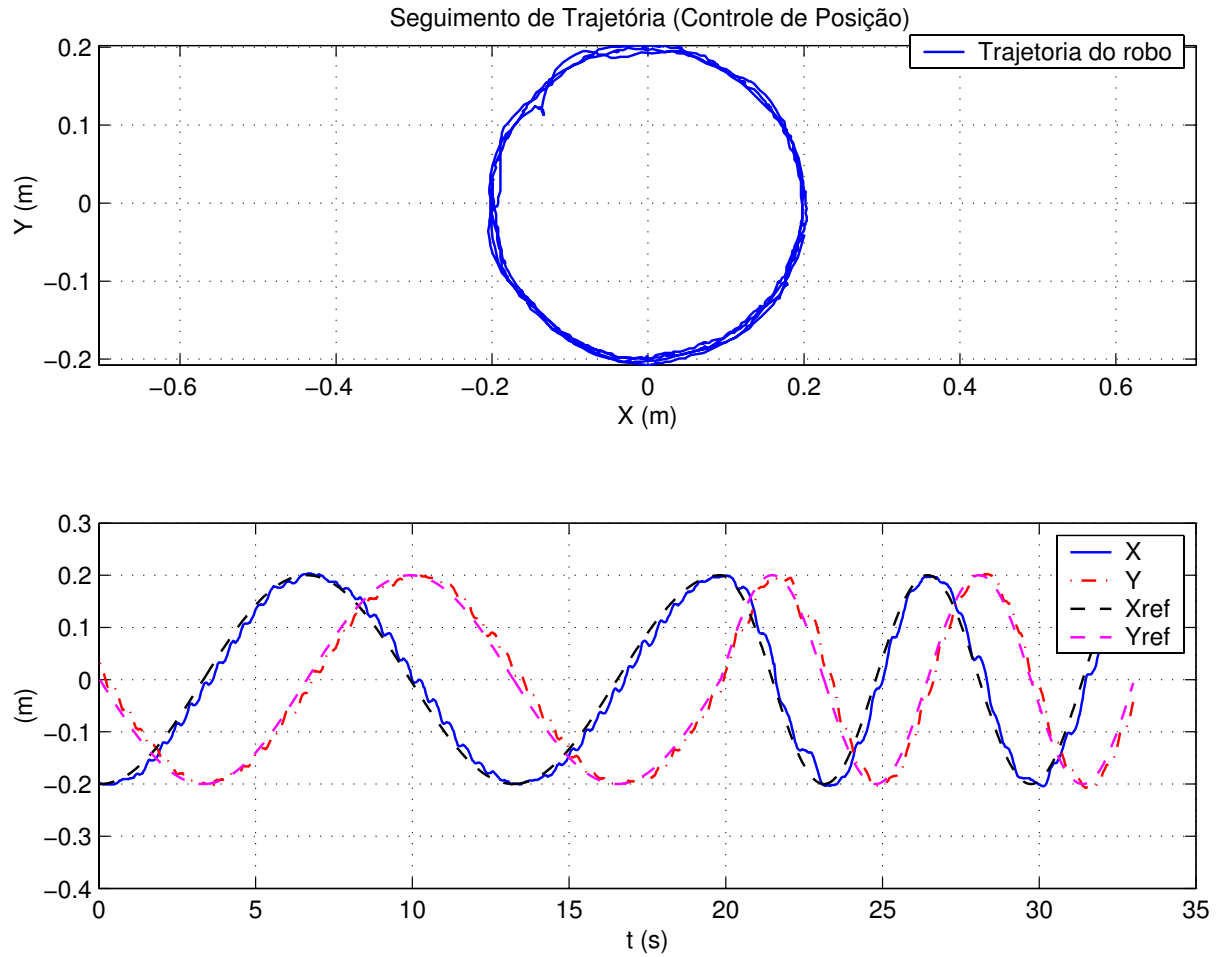


Figura 6.12: Resultados experimentais para a estratégia de controle de posição aplicada ao seguimento de caminho.

Na figura 6.12 o gráfico superior indica a trajetória percorrida pelo robô e o gráfico inferior apresenta a evolução das variáveis de controle (x e y) no tempo, assim como suas respectivas referências. Pode-se perceber que para atingir o objetivo de uma trajetória circular são passadas referências senoidais para x e y no espaço cartesiano.

6.2 Controle de Orientação

A seguir são apresentados resultados simulados e experimentais para o controle de orientação do robô. Assim como nas simulações geradas para o controle de posição, em cada figura de saída são apresentadas as evoluções das seguintes variáveis: sinal de controle, trajetória do robô no espaço cartesiano, erros linear e angular, e variáveis de configuração do robô (x , y e θ). Deve-se notar no gráfico que apresenta a trajetória do robô, a presença da referência móvel e sua evolução de acordo com a posição e orientação do robô. Serão abordadas simulações do controle de orientação para diferentes configurações finais. Para efeito de simplicidade, considera-se a mesma configuração inicial para os exemplos apresentados $q = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$, sem perda de generalidade, onde $q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$ e θ é dado em radianos. As simulações são apresentadas nas páginas que seguem com os devidos comentários.

Os resultados experimentais foram obtidos através de ensaios realizados na plataforma descrita no capítulo 3. Os resultados são apresentados e comentados nesta seção, demonstrando a eficácia da estratégia proposta. Para permitir uma melhor visualização dos resultados, foi dedicada uma página para cada simulação ou experimento.

- Configuração final $q = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^T$:

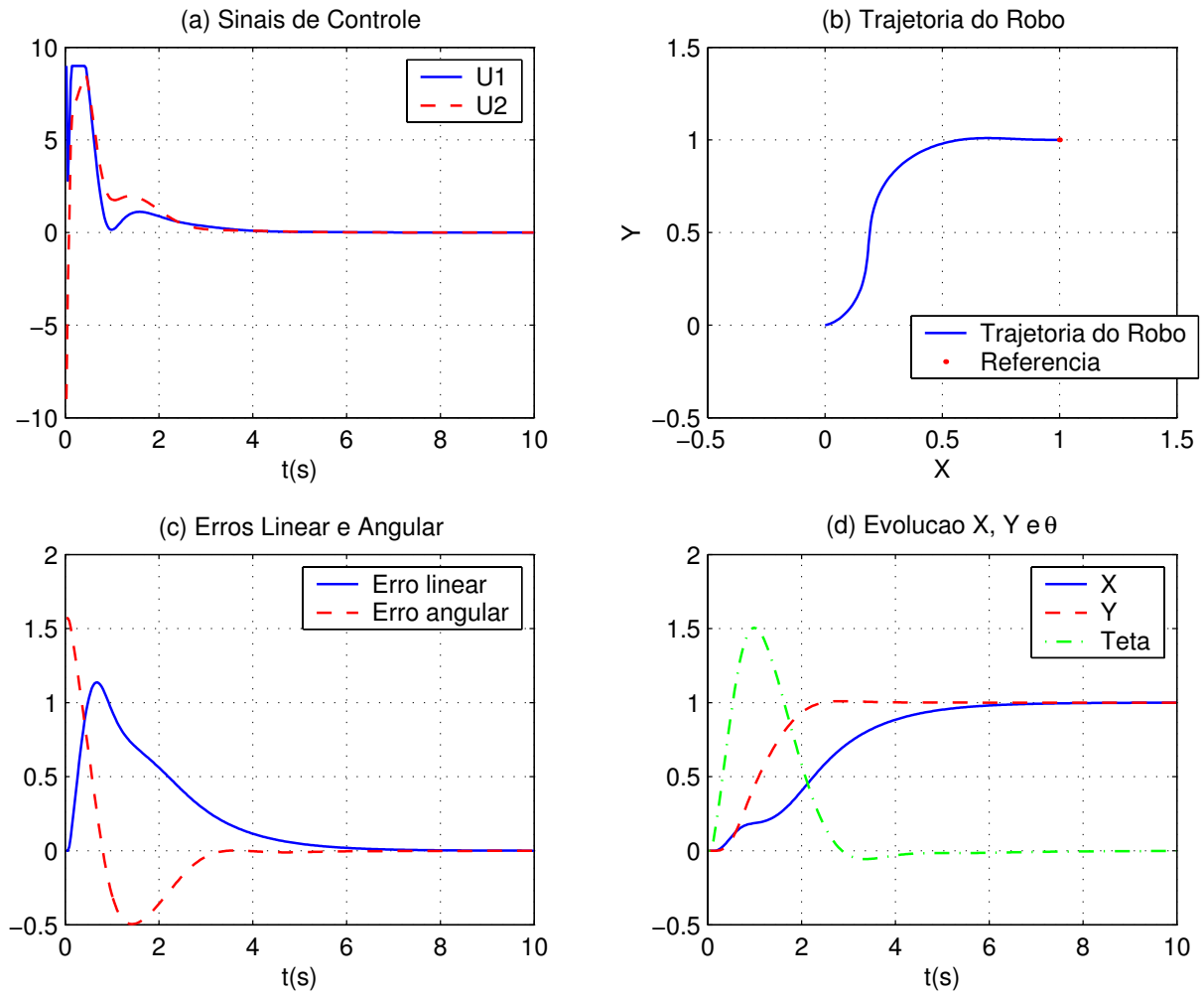


Figura 6.13: Controle de orientação - caso 1.

Pode-se observar na figura 6.13 que o sistema robótico atinge suavemente o ponto desejado $(x_d$ e y_d) no espaço cartesiano, assim como a orientação desejada θ_d . Como o controlador para dinâmica do sub-sistema linear foi projetado para sobre-sinal nulo, tem-se a aproximação suave do objetivo. Também pode-se perceber o comportamento da referência móvel em 6.13(b).

- Configuração final $q = \begin{bmatrix} 1 & 1 & \pi/2 \end{bmatrix}^T$:

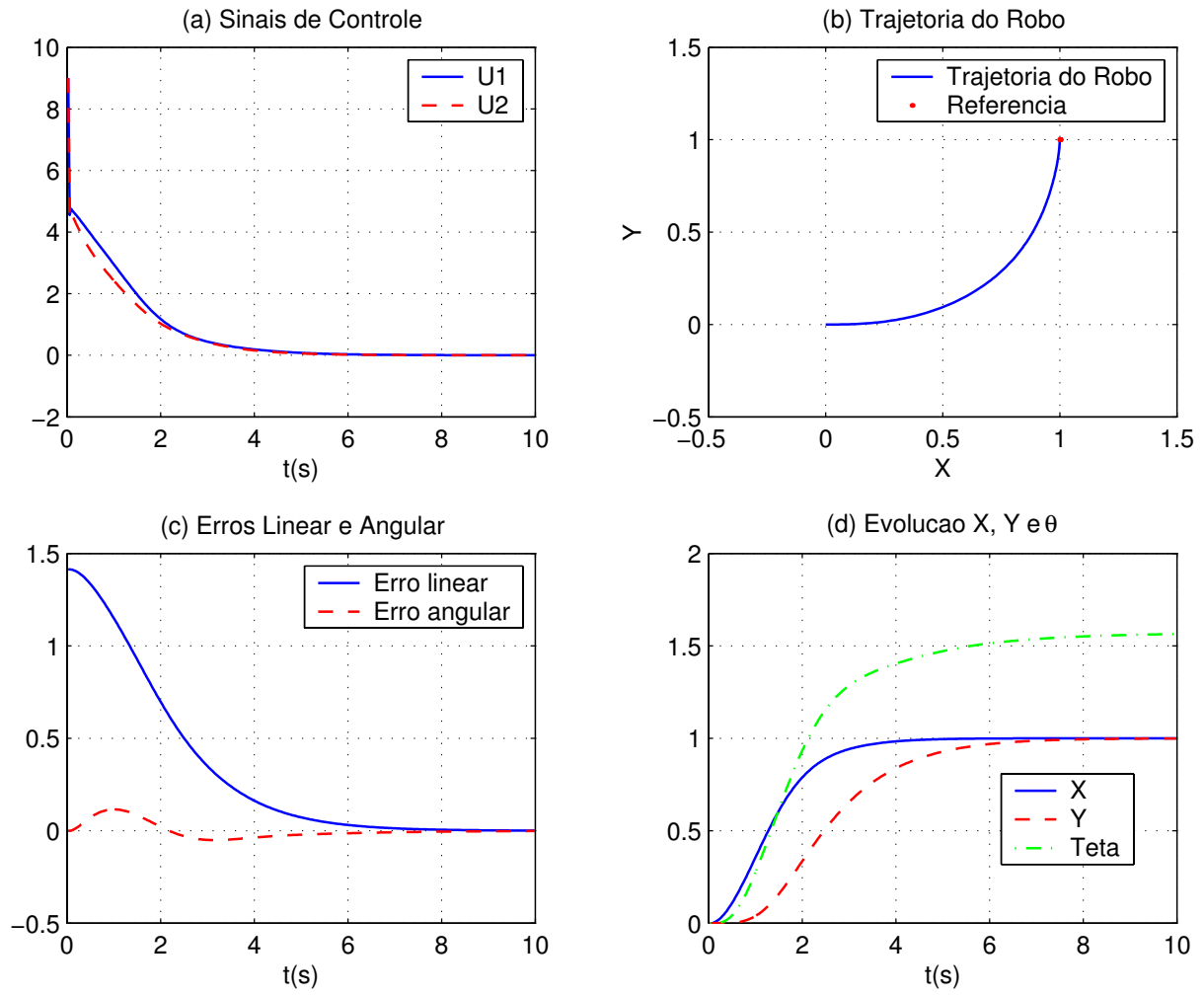


Figura 6.14: Controle de orientação - caso 2.

A figura 6.14 mostra que, para esta configuração, o sistema robótico atinge o ponto desejado $(x_d$ e y_d) no espaço cartesiano, assim como a orientação desejada θ_d , com sinais de controle suaves. No entanto, adiante serão apresentados casos onde ocorre o chaveamento da lei de controle.

- Configuração final $q = \begin{bmatrix} 1 & 1 & \pi \end{bmatrix}^T$:

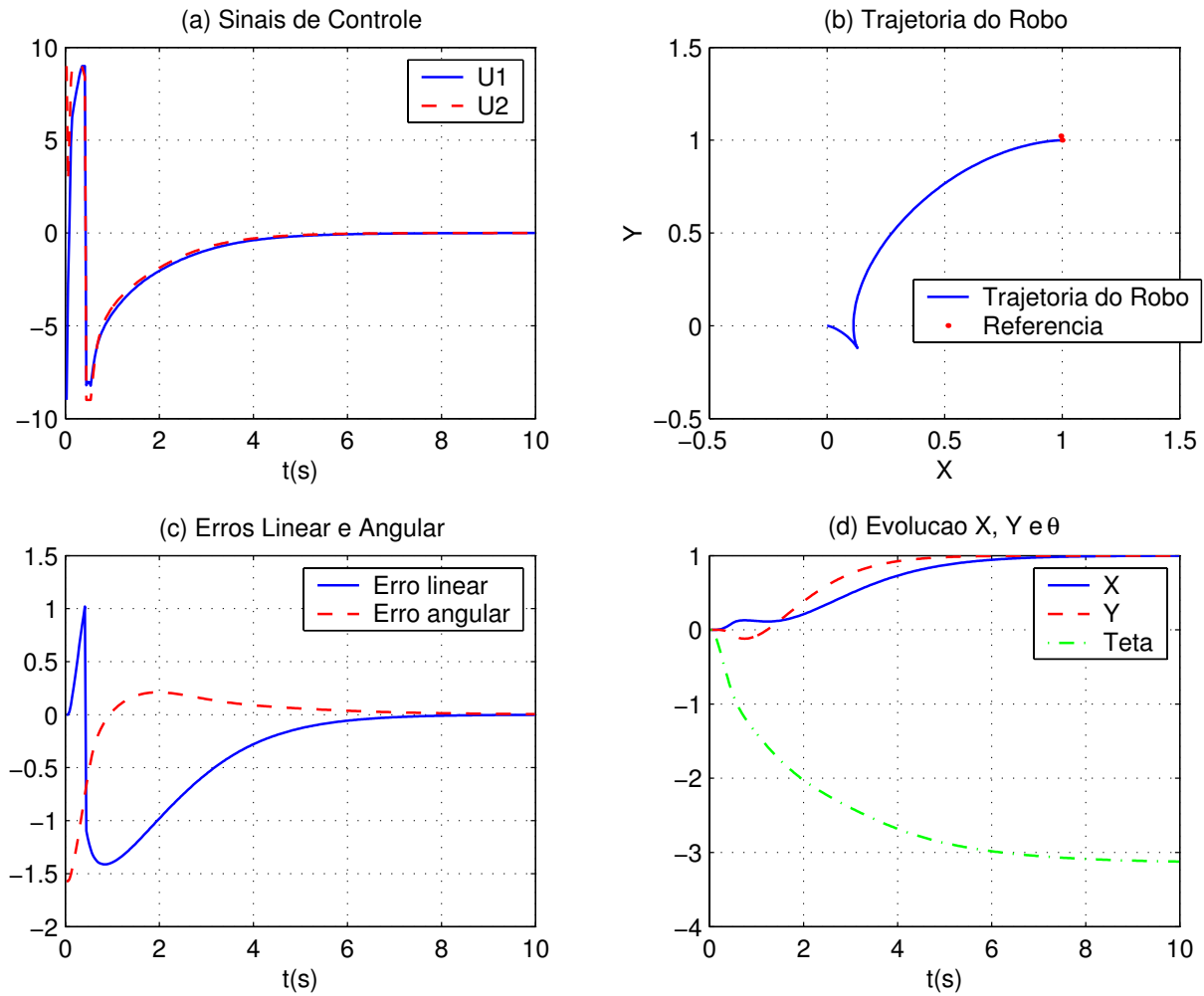


Figura 6.15: Controle de orientação - caso 3.

Neste caso o robô inicia o movimento linear, porém com maior ênfase à correção angular, visto que esta terá que sair de $\theta = 0$ para $\theta = \pi$. No entanto, ao atingir a condição $\Delta\phi > \pi/2$ a estratégia de controle permite ao robô detectar que o objetivo de controle encontra-se “atrás”, fazendo com que este se movimente com velocidade negativa até atingir o objetivo de controle de forma suave. Pode-se observar que a partir do chaveamento, as tensões u_1 e u_2 para os atuadores elétricos, tornam-se negativas, caracterizando velocidade negativa (o robô se movimenta para trás). Aqui fica claro que apesar de existirem chaveamentos na estratégia de controle, estes existem apenas para permitir correções no movimento linear do robô.

- Configuração final $q = \begin{bmatrix} 1 & 1 & -\pi/2 \end{bmatrix}^T$:

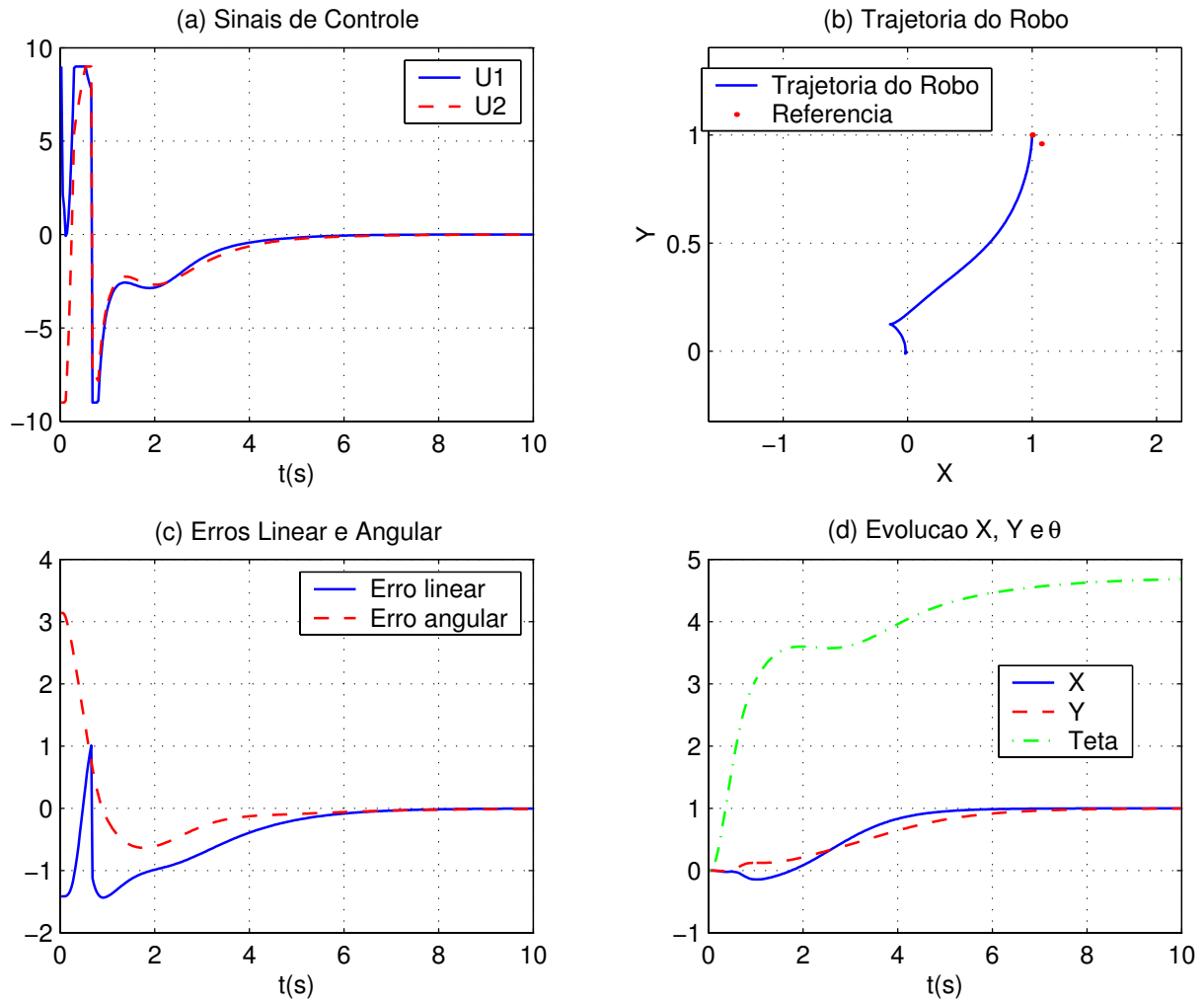


Figura 6.16: Controle de orientação - caso 4.

Neste caso o robô atinge o objetivo em três movimentos. O primeiro é um pequeno movimento para trás, conforme trecho de referência móvel gerado. O segundo é um movimento para frente procedendo correção angular, e por último um movimento para trás até atingir o objetivo. Neste exemplo pode-se observar que a referência móvel não é necessariamente contínua, permitindo ao robô flexibilidade para atingir o objetivo com a orientação desejada.

- Configuração final $q = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$:

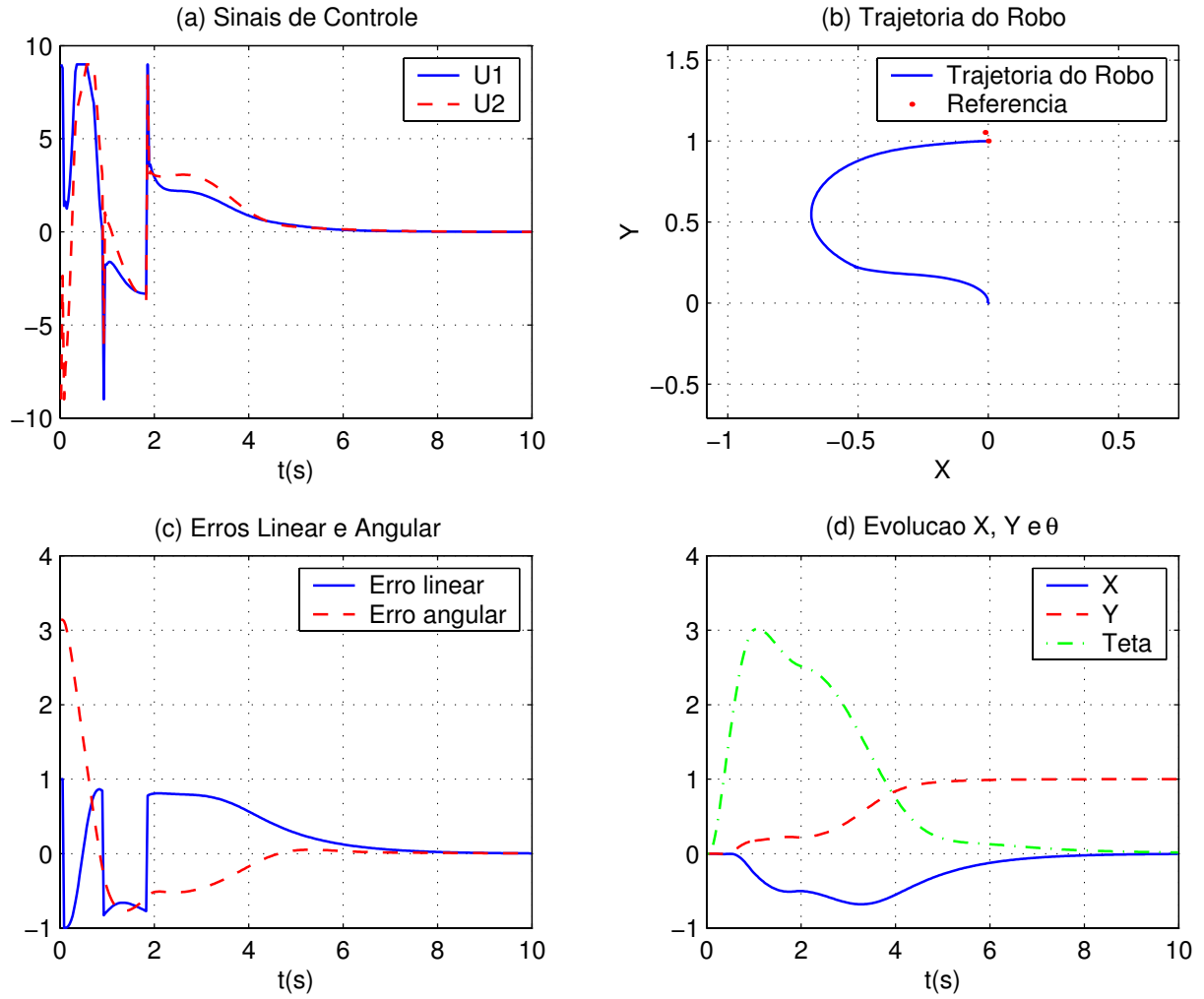


Figura 6.17: Controle de orientação - caso 5.

Neste caso tem-se uma configuração final onde $\theta = \theta_d$, $x = x_d$ e apenas y não atende à configuração desejada. Mesmo assim, a estratégia de controle gera referências adequadas que permitem a estabilização do robô em torno da configuração desejada, conforme figura 6.17(d). Pelas razões apresentadas, pode-se perceber que os sinais de controle também apresentam maior chaveamento, representando o esforço para atingir a configuração desejada.

- Configuração final $q = \begin{bmatrix} 0 & 1 & \pi/2 \end{bmatrix}^T$:

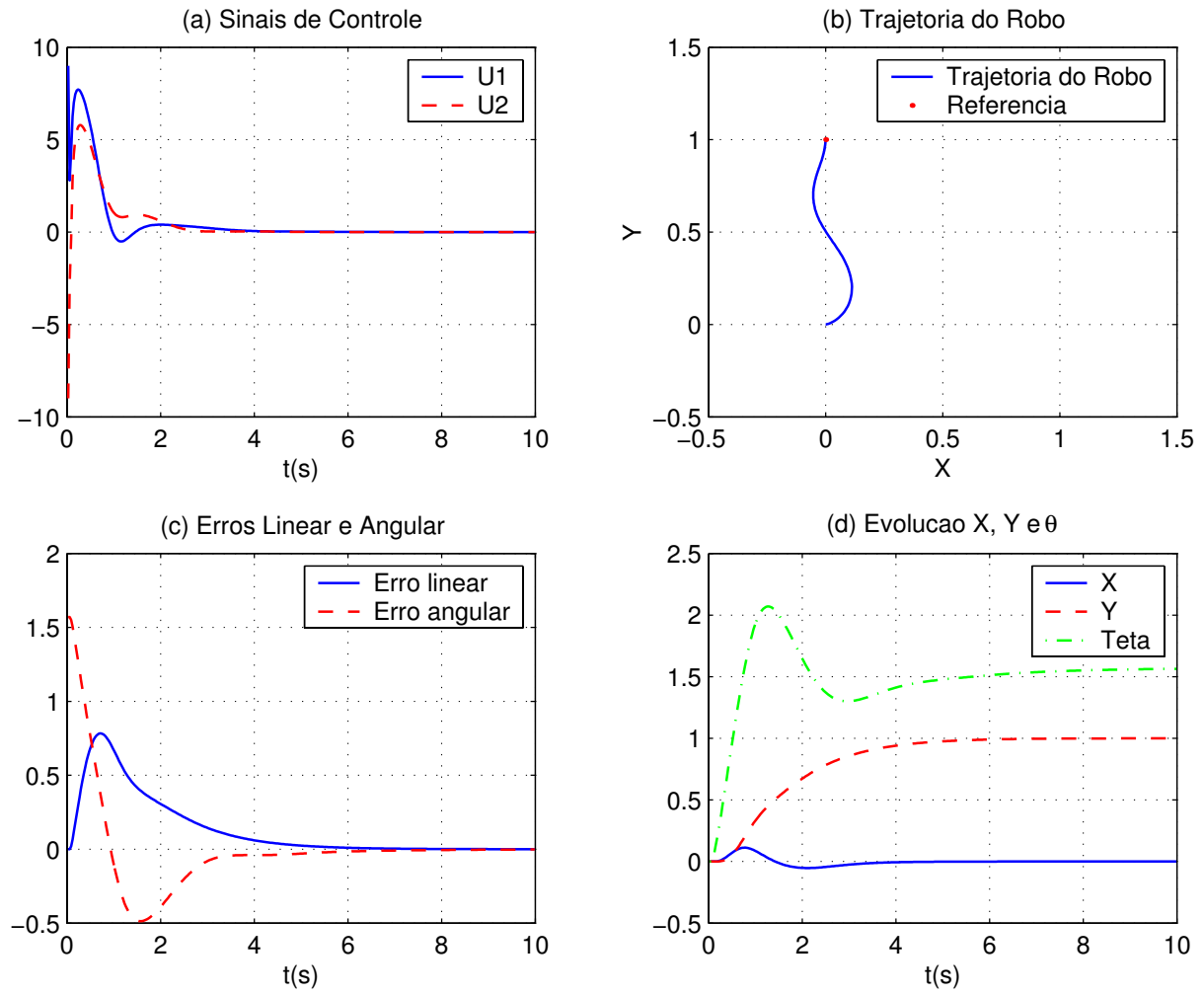


Figura 6.18: Controle de orientação - caso 6.

Neste caso tem-se que o robô atinge o objetivo com maior facilidade, sem a presença de chaveamentos no sinal de controle e no movimento do robô.

- Configuração final $q = \begin{bmatrix} -1 & 0 & \pi \end{bmatrix}^T$:

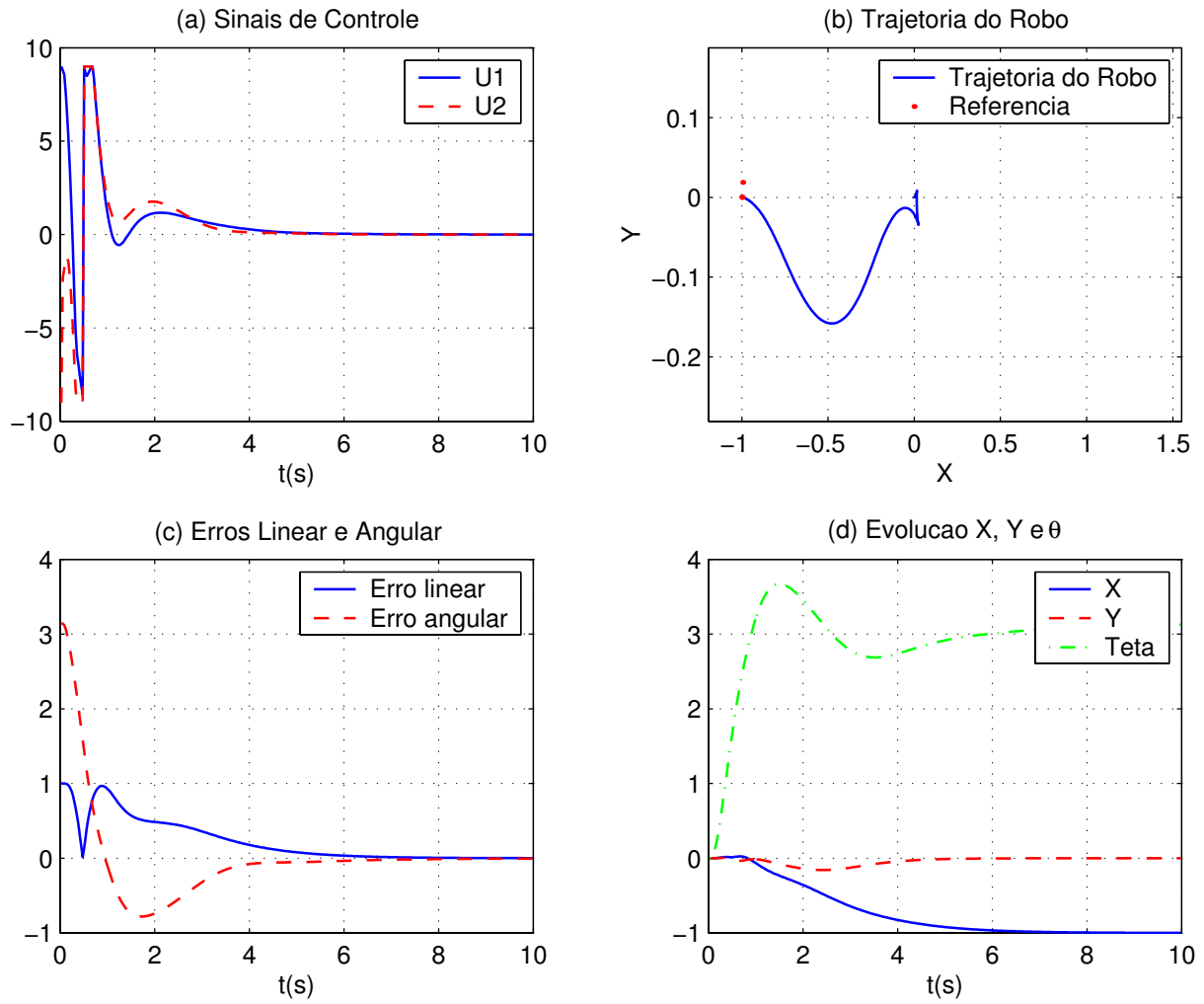


Figura 6.19: Controle de orientação - caso 7.

Este caso é similar ao apresentado na figura 6.16, isto é, o robô também realiza três movimentos para permitir o atingimento do objetivo com a orientação desejada.

- Configuração final $q = \begin{bmatrix} -1 & 0 & 0 \end{bmatrix}^T$:

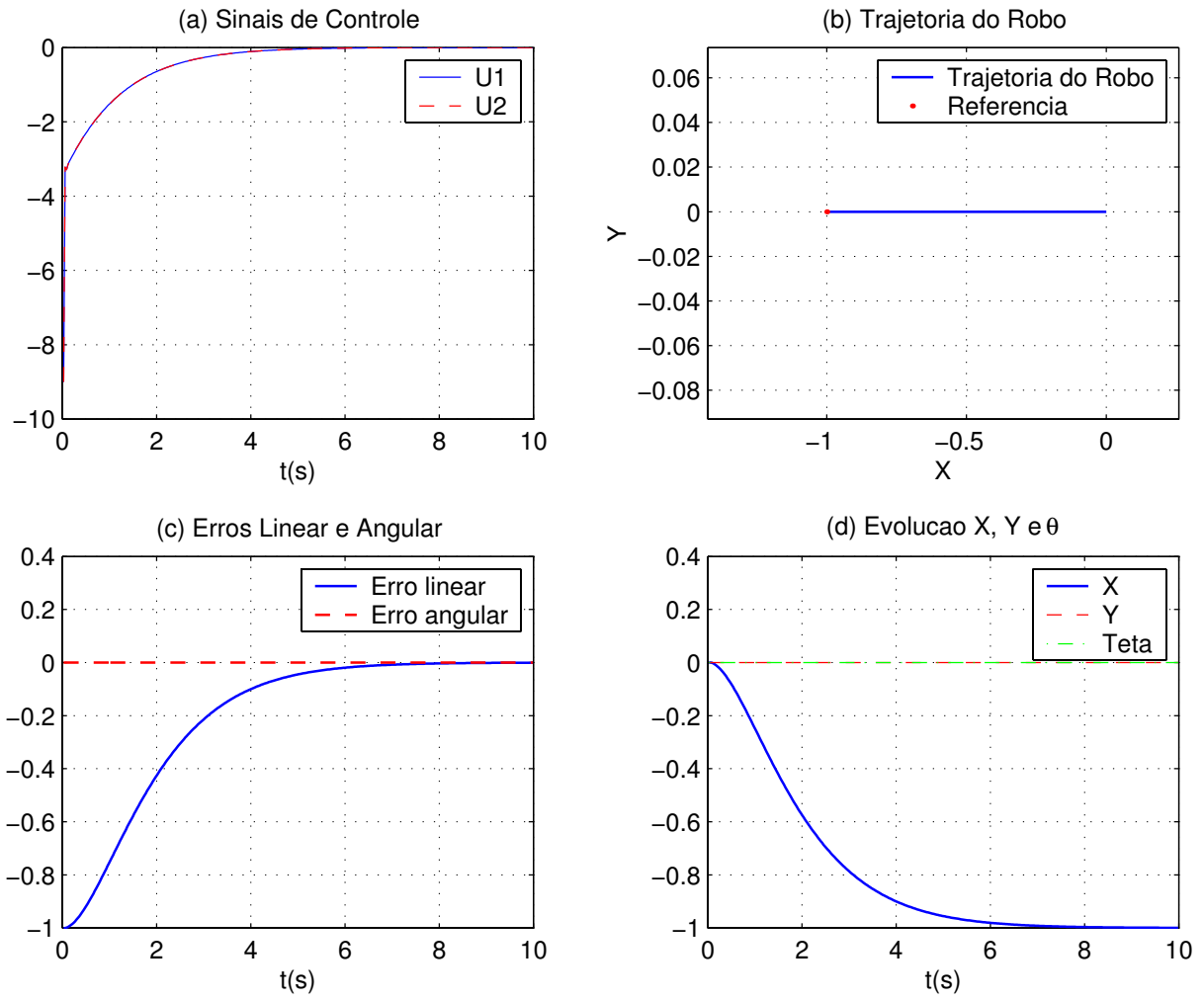


Figura 6.20: Controle de orientação - caso 8.

Este é um exemplo de um dos casos mais simples onde $y = y_d$, $\theta = \theta_d$ e apenas é necessário um movimento para trás para corrigir x , o que é realizado com suavidade.

- Configuração final $q = \begin{bmatrix} -1 & -1 & 0 \end{bmatrix}^T$:

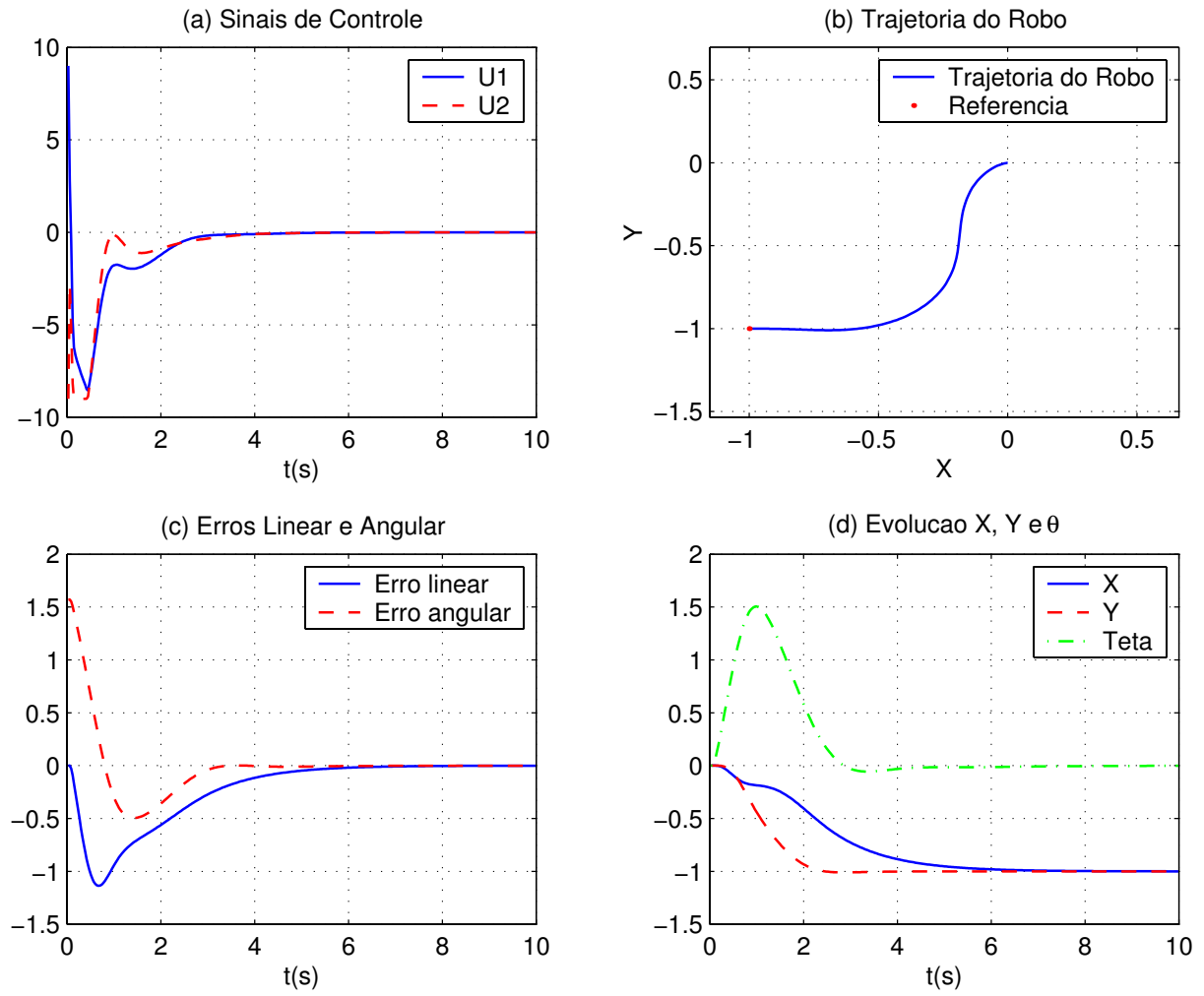


Figura 6.21: Controle de orientação - caso 9.

Neste exemplo mostra-se a capacidade do robô atingir o alvo com movimento para trás e realizando a correção angular necessária para atingir a configuração desejada.

- Configuração final $q = \begin{bmatrix} 1 & -1 & \pi/4 \end{bmatrix}^T$:

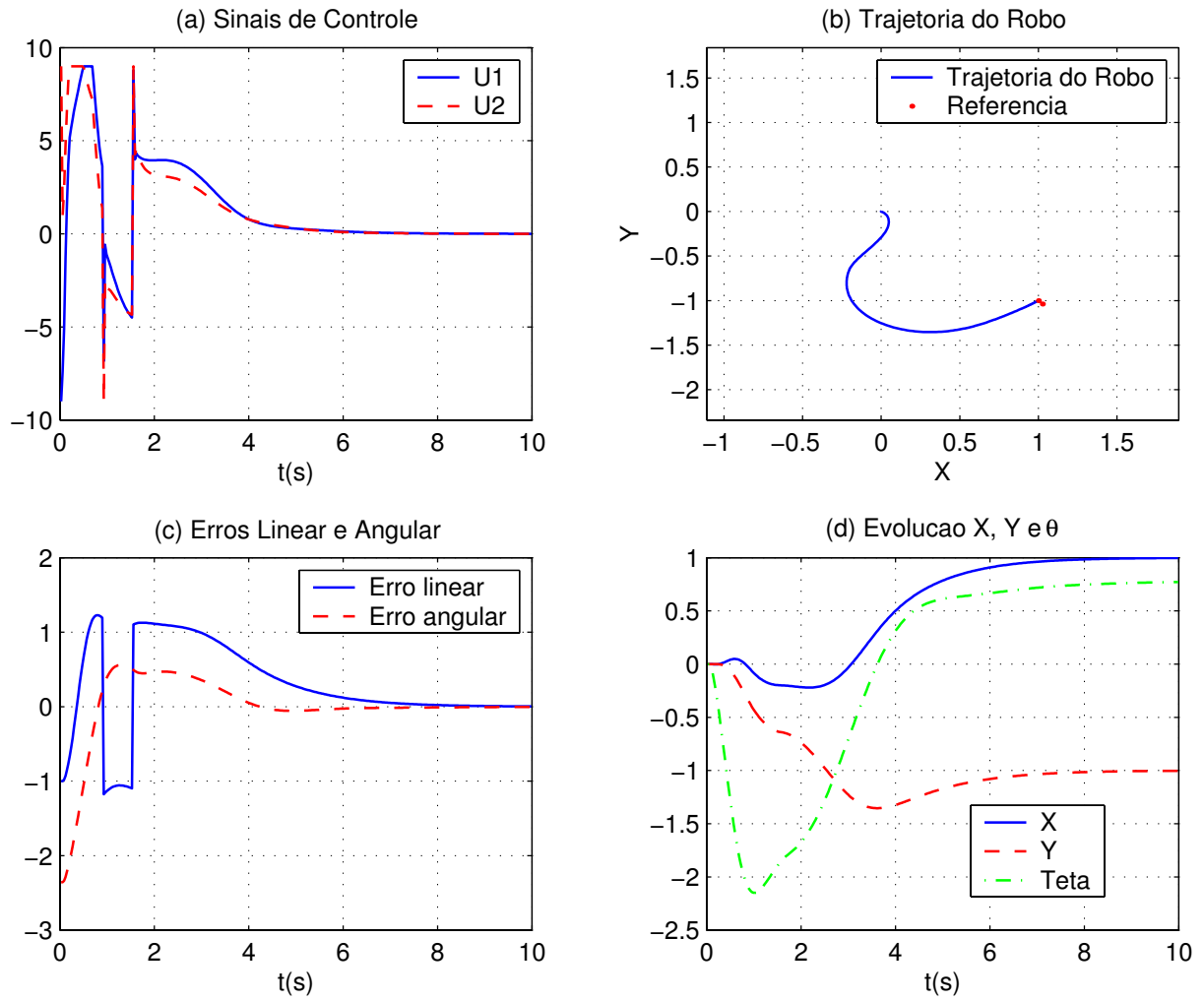


Figura 6.22: Controle de orientação - caso 10.

Sem comentários adicionais, apenas apresentando resultados para diferentes configurações finais.

- Configuração final $q = \begin{bmatrix} -1 & 1 & \pi/4 \end{bmatrix}^T$:

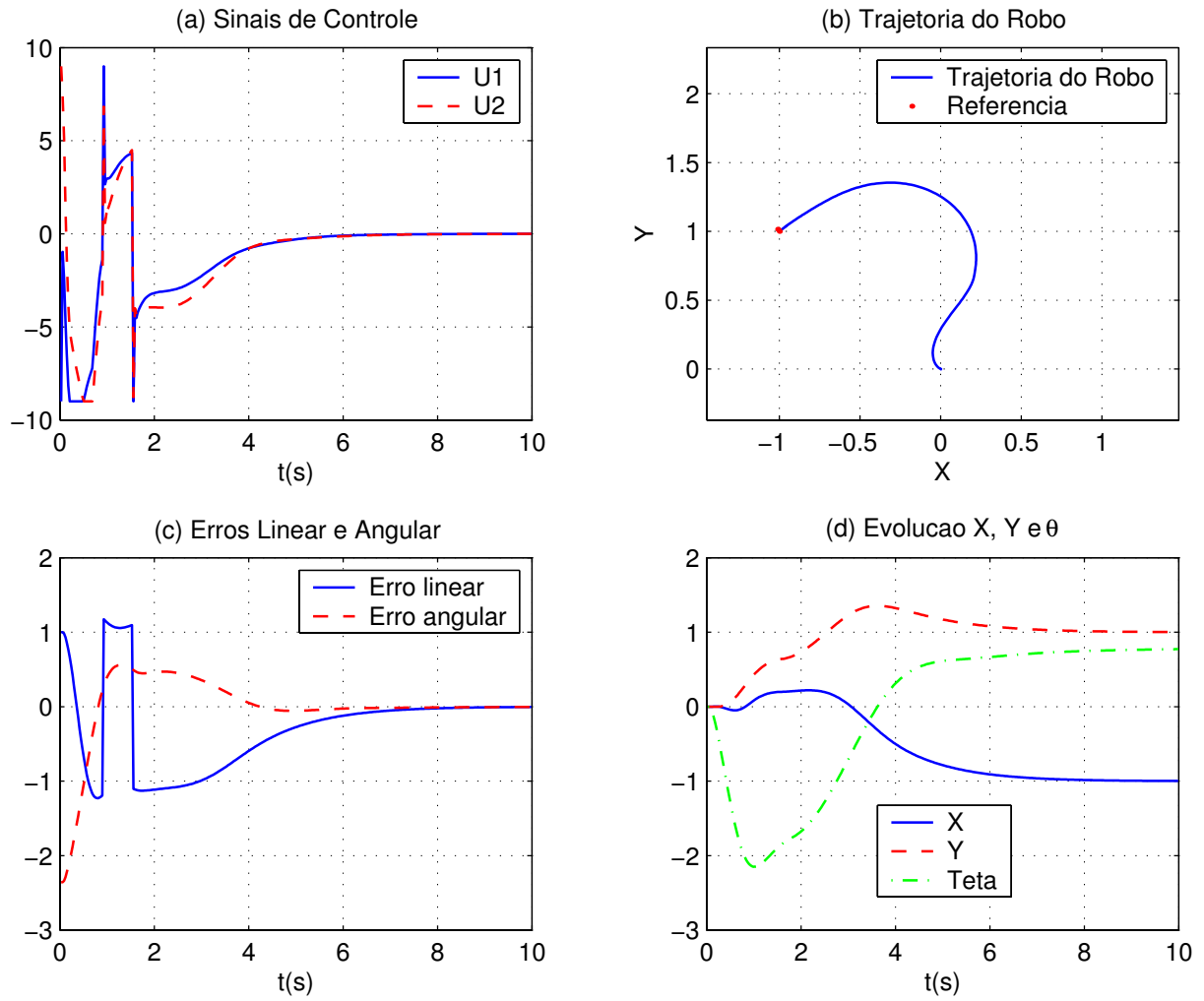


Figura 6.23: Controle de orientação - caso 11.

Sem comentários adicionais, apenas apresentando resultados para diferentes configurações finais.

6.2.1 Resultados Experimentais para Estratégia de Controle de Orientação

O primeiro resultado experimental apresentado trata de um ensaio envolvendo quatro configurações para estabilização do robô. As configurações $(x, y \text{ e } \theta)$ são representadas pelo conjunto $O_d = \{(0.0, -0.25, \pi/2); (-0.25, 0.0, 0), (0.0, 0.25, \pi/2), (0.55, 0.0, 0)\}$.

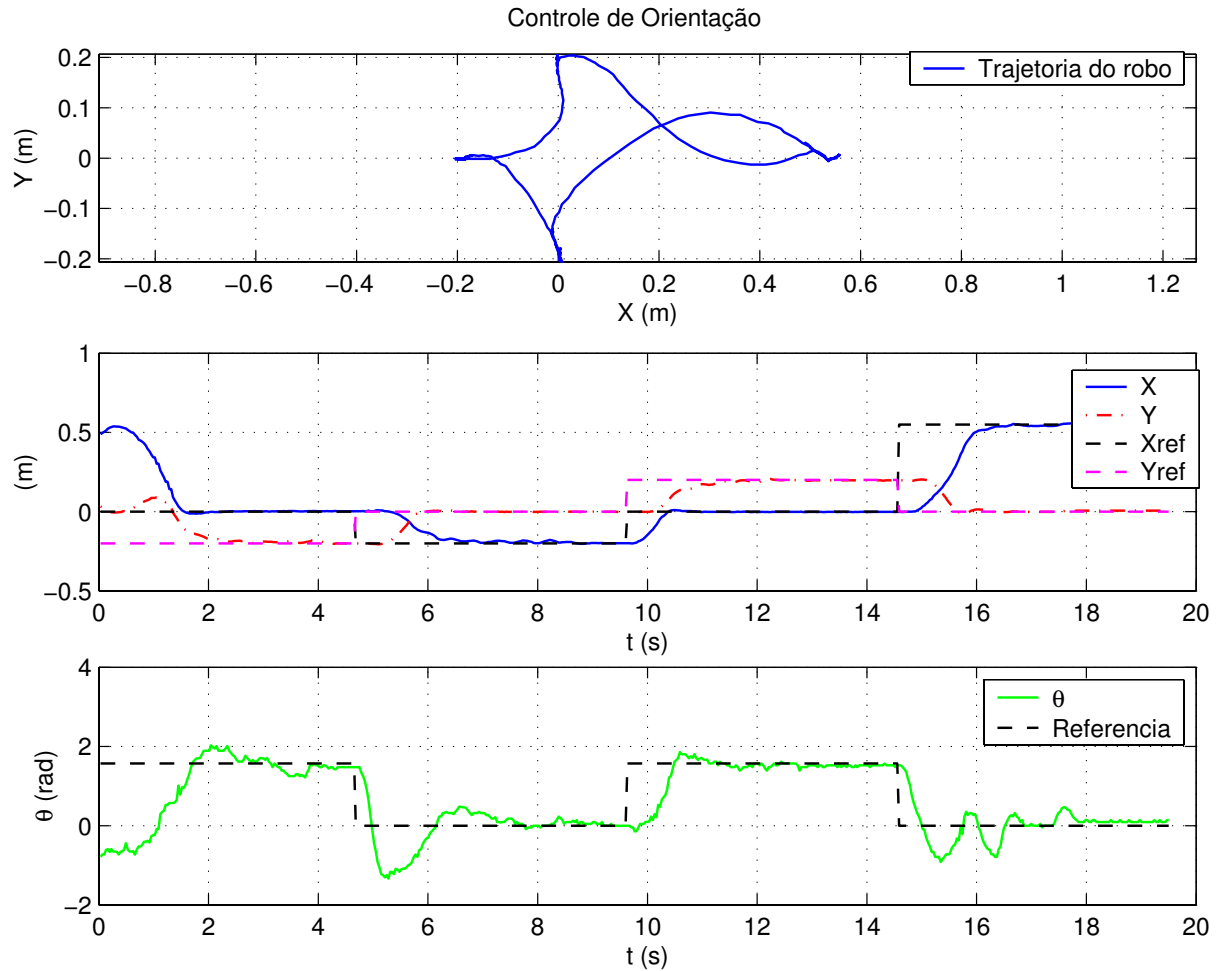


Figura 6.24: Resultados experimentais controle de orientação - caso 1.

Na figura 6.24 o gráfico superior indica a trajetória realizada pelo robô durante o experimento, o gráfico ao meio indica a evolução das variáveis de controle x e y , assim como suas respectivas referências, e o gráfico inferior indica a evolução da variável de controle θ e sua respectiva referência. É possível perceber através do resultado apresentado que a estratégia de controle de orientação cumpre seu objetivo, atingindo as referências de controle de acordo com o projeto.

O segundo resultado experimental apresentado explora a característica do controle multivariável exigido pelo sistema robótico. Neste ensaio as configurações $(x, y \text{ e } \theta)$ são representadas pelo conjunto $O_d = \{(-0.2, 0.0, 0); (0.2, 0.0, \pi/2), (-0.2, 0.0, 0)\}$. A figura 6.25 segue o mesmo modelo dos gráficos apresentados na figura 6.24.

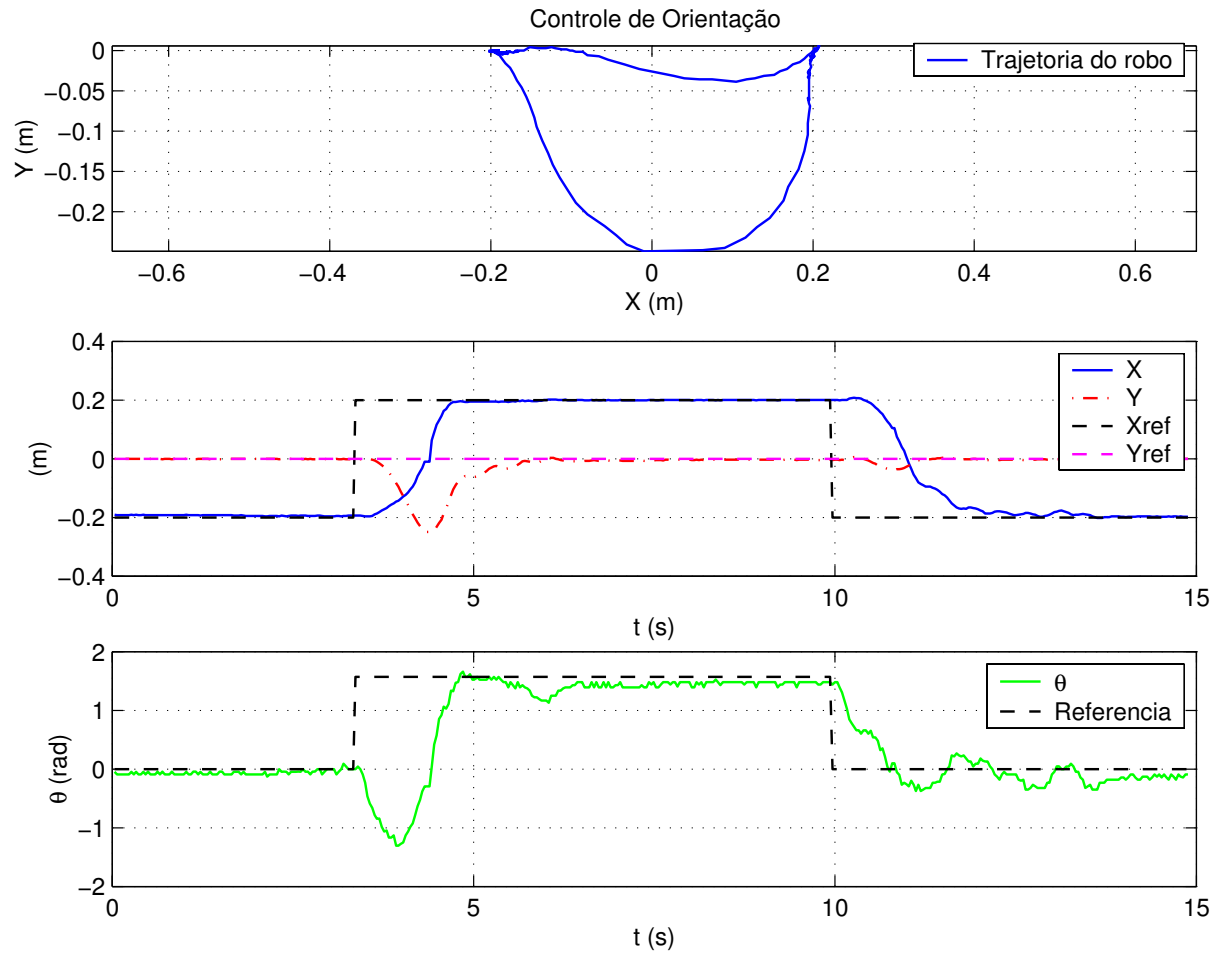


Figura 6.25: Resultados experimentais controle de orientação - caso 2.

É possível perceber nos resultados apresentados na figura 6.25 que da configuração inicial ($x_d = -0.2$, $y_d = 0$ e $\theta_d = 0$) para a configuração intermediária ($x_d = 0.2$, $y_d = 0$ e $\theta_d = \pi/2$) não há erro na coordenada y , no entanto, a estratégia de controle permite uma piora desta variável do sistema a fim de que seja alcançado o objetivo de controle, o que é uma característica de sistemas de controle multivariáveis.

O erro em regime apresentado na resposta da referência angular deve-se principalmente à não-linearidades não modeladas como, por exemplo, o efeito do atrito seco, que impacta fortemente o desempenho do sistema robótico utilizado nos experimentos.

O terceiro resultado experimental para o controle de orientação apresenta o efeito do controle multivariável, agora considerando fixa a referência para a orientação do robô. Neste ensaio as configurações para o robô (x , y e θ) são representadas pelo conjunto $O_d = \{(0.1, 0.1, 0); (-0.1, -0.1, 0)\}$. A figura 6.26 segue o mesmo modelo dos gráficos apresentados nas figuras 6.24 e 6.25.

É importante verificar que, como no caso anterior, mesmo com variáveis de controle apresentando erro nulo, o sistema de controle força uma situação de desequilíbrio desta variável para

permitir a estabilização em torno do ponto de equilíbrio desejado (nova configuração). Pode-se notar novamente o efeito devido às não-linearidades não modeladas do sistema robótico que impactam diretamente em um erro em regime na resposta do sistema, principalmente com relação à orientação.

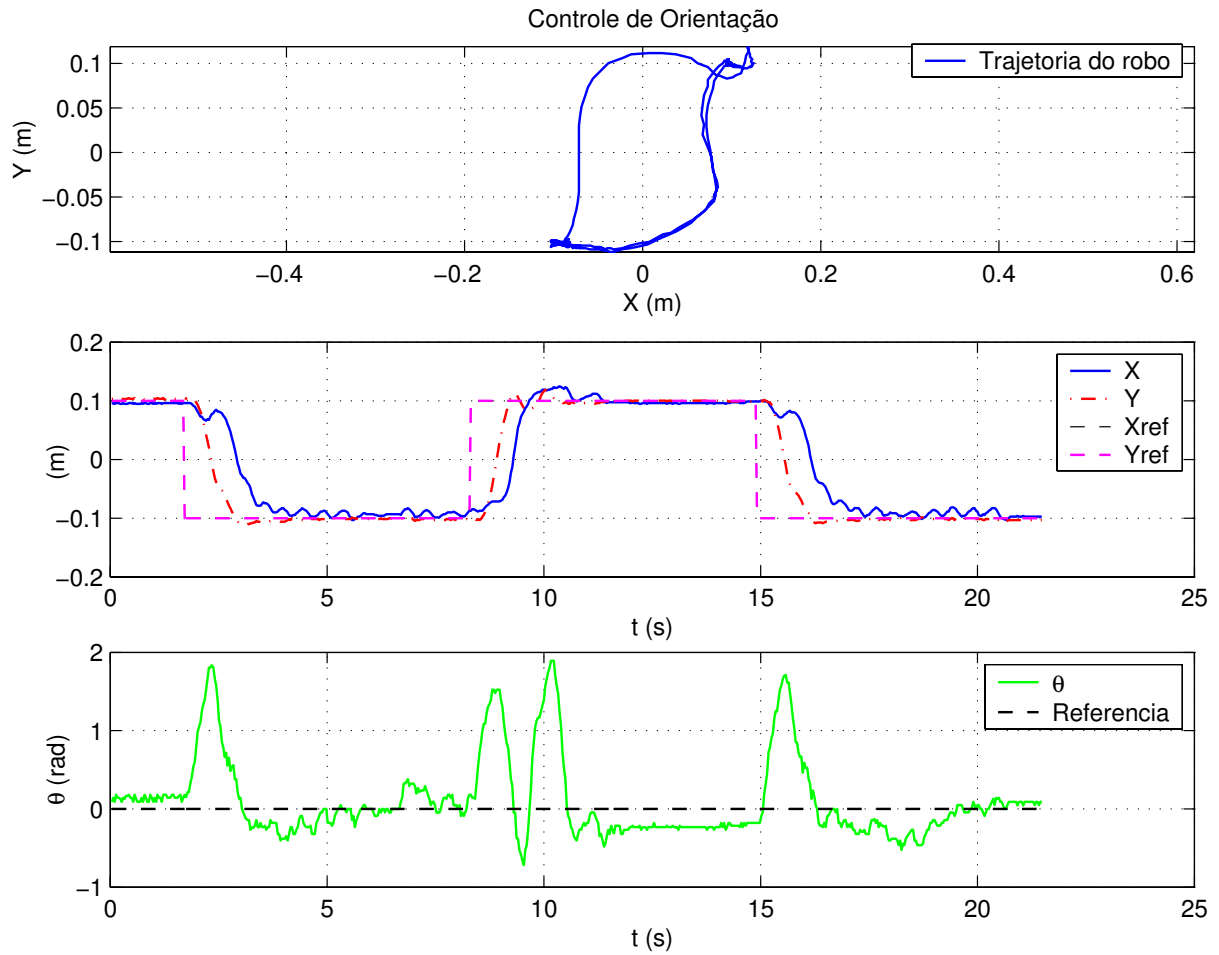


Figura 6.26: Resultados experimentais controle de orientação - caso 3.

Os resultados experimentais apresentados demonstram a capacidade da estratégia de controle de lidar com o problema de estabilização de robôs móveis não-holonômicos com acionamento diferencial.

6.3 Conclusão

Os resultados apresentados mostram a flexibilidade das estratégias desenvolvidas para controle de posição e orientação em lidar com diferentes configurações iniciais e finais. Deve-se lembrar que as estratégias são válidas apenas para $\Delta l > 0$, e implementações práticas podem incluir esta condição. Isto quer dizer que há uma descontinuidade na origem ($\Delta l = 0$) que deve ser tratada na implementação prática da estratégia de controle de acordo com os requisitos das tarefas a serem executadas pelo sistema robótico.

Foi demonstrada a capacidade da estratégia de controle de posição ser acoplada a um gerador de trajetórias, incluindo resultados experimentais que comprovam sua eficácia.

A partir de condições de ruído e incertezas geradas pelo simulador, percebeu-se que, apesar de haver uma interferência no sinal de controle, o desempenho do sistema não apresentou degradação visível. Detectou-se que o efeito que mais degrada o sinal de controle é o efeito de “pixelização”, pois com este as informações utilizadas pelo controlador são quantizadas, o que gera incertezas para o sistema. No entanto, comparando-se os resultados simulados apresentados aos resultados experimentais, percebe-se que estes problemas são contornáveis e é possível obter um bom desempenho do sistema de controle.

Os resultados experimentais apresentados demonstram a aplicabilidade das estratégias desenvolvidas para a classe de robôs considerada neste trabalho. Além disso, este trabalho abre espaço para utilização de outras técnicas de controle, diferentes de controladores PID, para otimização do desempenho do sistema robótico, até mesmo incluindo a determinação em tempo real dos parâmetros referentes à dinâmica do robô.

Capítulo 7

Considerações Finais

Neste trabalho foram introduzidos os principais conceitos de robótica móvel e sua interface com a área de sistemas de controle. O problema de controle para sistemas robóticos não-holonômicos com rodas foi abordado e as principais estratégias propostas na literatura foram estudadas. No estudo bibliográfico foi possível notar que há uma deficiência de trabalhos relativos à compensação dinâmica de robôs móveis não-holonômicos e na maioria dos casos as estratégias são complexas ou difíceis de reproduzir (como é o caso das implementações através de redes neurais artificiais).

Mostrou-se que as estratégias de controle de posição e orientação desenvolvidas apresentam resultados experimentais de acordo com a proposta inicial deste trabalho. Buscou-se no decorrer do desenvolvimento chegar a uma estratégia de controle de fácil implementação. Este objetivo foi atingido, o que permite ampla utilização das estratégias propostas em sistemas robóticos não-holonômicos com acionamento diferencial. Outra vantagem é a possibilidade de utilizar controladores dinâmicos lineares e em casos específicos, podem ser utilizados controladores monovariáveis (tipo PID), conforme apresentado neste trabalho.

Para testar as estratégias de controle durante o desenvolvimento deste trabalho foi implementado um simulador incorporando o modelo dinâmico de um sistema robótico com as características do robô real utilizado para os experimentos. Este simulador buscou integrar a maioria dos efeitos que podem afetar diretamente o desempenho do sistema de controle.

No decorrer deste trabalho foram comentados aspectos de estabilidade da estratégia de controle com base apenas em observações do comportamento do sistema para diferentes características do sistema de controle. No entanto, não há ainda uma prova formal de estabilidade para o sistema de controle proposto. Isto deverá ser investigado em trabalhos futuros.

Outro aspecto a ser abordado em trabalhos futuros é abrangência desta estratégia em con-

junto com técnicas de controle para sistemas multivariáveis, como *model matching* e GPC (*Generalized Predictive Controller*), a fim de verificar a resposta do sistema de controle quando existe um grau de acoplamento elevado entre as dinâmicas dos movimentos angular e linear do sistema robótico.

Uma desvantagem da estratégia proposta é a necessidade do conhecimento dos parâmetros dinâmicos do robô, que não são de fácil medição. Este ponto será contornado através da utilização de um identificador *online* de parâmetros dinâmicos para robôs móveis com acionamento diferencial, desenvolvido por Guerra et. al, 2004 [8]. A partir deste sistema será possível acoplá-lo às estratégias desenvolvidas de forma a atingir um sistema de controle para robôs móveis mais completo.

Os resultados experimentais apresentados demonstram a aplicabilidade das estratégias desenvolvidas para a classe de robôs considerada, atingindo de forma concreta os objetivos apontados no início deste trabalho e cumprindo com as premissas básicas de realizar o controle dinâmico baseado apenas na configuração do robô.

Trabalhos publicados a partir dos estudos realizados durante o desenvolvimento desta dissertação (Vieira et. al, 2004 [21] e Vieira et. al, 2003 [20]) são citados nas referências bibliográficas.

Referências Bibliográficas

- [1] M. Aicardi, G. Casalino, A. Bicchi, and A. Balestrino. Closed loop steering of unicycle-like vehicles via lyapunov techniques. *IEEE R&A Magazine*, vol. 2, no. 1, pp.27–35, 1995.
- [2] K. R. T. Aires, P. J. Alsina, and A. A. D. Medeiros. A global vision system for mobile mini-robots. *Anais do V Simpósio Brasileiro de Automação Inteligente*, Brasil, 2001.
- [3] R. W. Brockett, R. S. Millman, and H. J. Sussmann. Asymptotic stability and feedback stabilization. *Differential Geometric Control Theory*, Birkhäuser, Boston, USA, pp. 181–191, 1983.
- [4] E. F. Camacho and C. Bordons. *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. Springer, 2nd edition, London, 2000.
- [5] B. d’Andrea Novel, G. Bastin, and G. Campion. Control of nonholonomic wheeled mobile robots by state feedback linearization. *Int. Journal of Robotics Research*, vol. 14, no. 6, pp. 543–559, 1995.
- [6] L. C. Figueiredo and F. Jota. Introdução ao controle de sistemas não-holonômicos. *Revista Controle e Automação*, vol. 15, no. 3, pp. 243–268, 2004.
- [7] P. Gaudiano, E. Zalama, and J. L. Coronada. An unsupervised neural network for low-level control of a wheeled mobile robot: noise resistance, stability and hardware implementation. *Unknown resource*, 1995.
- [8] P. N. Guerra, P. J. Alsina, A. A. D. Medeiros, and A. P. Araújo Jr. Linear modelling and identification of a mobile robot with differential drive. *I ICINCO - International Conference on Informatics in Control, Automation and Robotics.*, Setúbal, Portugal, 08/2004.
- [9] W. F. Lages and E. M. Hemerly. Controle de robôs móveis utilizando transformação descontínua e linearização adaptativa. *Anais do CBA 2000*, Florianópolis-SC, Brasil, Setembro, 2000.

- [10] F. C. Lizarralde. *Stabilization of Affine Nonlinear Control Systems by a Newton Type Method*. PhD thesis, Universidade Federal do Rio de Janeiro, COPPE, Rio de Janeiro – Brasil, 1998.
- [11] A. De Luca, G. Oriolo, L. Paone, and P. R. Giordano. Experiments in visual feedback control of a wheeled mobile robot. *Proc. IEEE Int. Conf. on Robotics & Automation*, Washington–DC, USA, May, 2002.
- [12] A. De Luca, G. Oriolo, C. Samson, and J. P. Laumond. *Robot Motion Planning and Control*, chapter Feedback Control of a Nonholomic Car-like Robot. Lectures Notes in Control and Information Sciences 229. Springer, 1998.
- [13] K. Ogata. *Modern Control Engineering*. Prentice Hall, 3rd edition, 1997.
- [14] V. M. de Oliveira and W. F. Lages. Controle em malha fechada de robôs móveis utilizando redes neurais e transformação descontínua. *Anais do V Simpósio Brasileiro de Automação Inteligente*, Brasil, Novembro, 2001.
- [15] G. A. S. Pereira. Identificação e controle de microrrobôs móveis. Master’s thesis, Universidade Federal de Minas Gerais, PPGEE, Belo Horizonte, 2000.
- [16] G. A. S. Pereira, M. F. M. Campos, and L. A. Aguirre. Modelo dinâmico para predição da posição e orientação de micro-robôs móveis. *Anais do XIII Congresso Brasileiro de Automática*, Brasil, Setembro, 2000.
- [17] C. Samson. Time-varying feedback stabilization of car-like wheeled mobile robots. *Int. Journal of Robotics Research*, vol. 12, no. 1, pp. 55–64, 1993.
- [18] H. G. Tanner and K. J. Kyriakopoulos. Discontinuous backstepping for stabilization of nonholomic mobile robots. *Proc. of the 2002 IEEE Int. Conf. on Robotics & Automation*, Washington–DC, May, 2002.
- [19] F. C. Vieira, P. J. Alsina, and A. A. D. Medeiros. Micro-robot soccer team - mechanical and hardware implementation. *Anais do XVI Congresso Brasileiro de Engenharia Mecânica*, Brasil, Novembro, 2001, pp. 534-540.
- [20] F. C. Vieira, A. A. D. Medeiros, and P. J. Alsina. Dynamic stabilization of a two-wheeled differentially driven nonholonomic mobile robot. *SBAI 2003 - Simpósio Brasileiro de Automação Inteligente*, Bauru, SP, Brazil, 09/2003, pp. 620-624.

- [21] F. C. Vieira, A. A. D. Medeiros, P. J. Alsina, and A. P. Araújo Jr. Position and orientation control of a two-wheeled differentially driven nonholonomic mobile robot. *I ICINCO - International Conference on Informatics in Control, Automation and Robotics.*, Setúbal, Portugal, 08/2004.
- [22] M. M. Yamamoto, D. P. F. Pedrosa, and A. A. D. Medeiros. Um simulador dinâmico para mini-robôs móveis com modelagem de colisões. *Anais do VI Simpósio Brasileiro de Automação Inteligente*, Bauru, SP, Brasil, Setembro 2003.
- [23] J. M. Yang and J. H. Kim. Sliding mode motion control of nonholonomic mobile robots. *IEEE Control Systems Magazine*, pp.15–23, April, 1999.