

Universidade Federal do Rio Grande do Norte
Centro de Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica

Diogo Pinheiro Fernandes Pedrosa

**Sistema de Navegação para Robôs Móveis
Autônomos**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Rio Grande do Norte, como parte dos requisitos necessários para obtenção do grau de Mestre em Ciências (M.Sc.).

Orientador: Prof D.Sc. Adelardo Adelino Dantas de Medeiros
Co-orientador: Prof D.Sc. Pablo Javier Alsina

Natal/RN - Brasil
Agosto de 2001

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Agradecimentos

Primeiramente agradeço a Deus por esta bênção. Seria bastante difícil concluir tal trabalho sem um equilíbrio espiritual.

Agradeço aos meus queridos Adelardo e Pablo pela orientação, paciência e encorajamento. Agradeço aos colegas de mestrado Kelson e Rodrigo pela companhia e pela disposição de abraçar um tema até então inédito na UFRN que é o futebol de robôs. Agradeço aos alunos de iniciação científica que prestaram toda uma acessoria para este trabalho: Frederico, Yamamoto, Patrícia e outros que vieram e ainda virão para nos ajudar.

Agradecimentos mais que especiais para os meus familiares. Aos meus pais, Oliveira e Lutigardes, e irmãos, Alan, Alex e Igor, com muito carinho dedico esta vitória. Aos meus sogros, Tadeu e Socorro, e cunhados, Júnior e Tatiany, obrigado pela “força”. Porém, com muito amor agradeço à minha esposa, Alaíde Tereza, mulher de enorme coração, e ao nosso filho, Emanuel, que verdadeiramente é meu grande trabalho de mestrado. Obrigado a todos vocês.

Sumário

| | |
|---|------------|
| Resumo | vi |
| Abstract | vii |
| 1 Introdução | 1 |
| 1.1 Objetivos | 3 |
| 1.2 Estrutura da Dissertação | 4 |
| 2 Futebol de Robôs | 5 |
| 2.1 Características Gerais | 5 |
| 2.1.1 O Futebol da FIRA | 6 |
| 2.2 O Futebol de Robôs do DCA-UFRN | 7 |
| 2.3 Conclusões | 8 |
| 3 O Problema da Navegação | 9 |
| 3.1 Navegação de Robôs | 9 |
| 3.2 Planejamento e Geração do Movimento | 10 |
| 3.2.1 Restrições Não-Holonômicas | 12 |
| 3.3 Controle do Movimento | 14 |
| 3.4 Técnicas Utilizadas no Futebol de Robôs | 14 |
| 3.5 Propostas para a Equipe do DCA-UFRN | 15 |
| 3.6 Conclusões | 16 |
| 4 Estimação da Configuração dos Minirobôs | 17 |
| 4.1 Introdução | 17 |
| 4.2 Algoritmo dos Mínimos Quadrados | 18 |
| 4.3 Aplicação no Futebol de Robôs | 19 |
| 4.3.1 Normalização do Ângulo | 21 |
| 4.4 Conclusões | 22 |
| 5 Planejamento de Caminhos | 25 |
| 5.1 Motivação | 25 |
| 5.2 Abordagem Clássica | 26 |
| 5.3 Aproximação por Elipse | 26 |
| 5.4 Polinômios Paramétricos | 28 |
| 5.4.1 Polinômios de Segundo Grau | 28 |
| 5.4.2 Polinômios de Terceiro Grau | 29 |
| 5.5 Adequação dos Coeficientes Livres | 32 |

| | | |
|----------|---|-----------|
| 5.6 | Geração da Trajetória | 34 |
| 5.7 | Conclusões | 36 |
| 6 | Resultados Experimentais | 38 |
| 6.1 | Modelagem Matemática do Minirobô | 38 |
| 6.1.1 | Esforços sobre o Robô | 40 |
| 6.2 | Controle dos Minirobôs | 43 |
| 6.2.1 | Estratégia de Controle | 43 |
| 6.3 | Simulador dos Minirobôs | 46 |
| 6.4 | Resultados Experimentais | 47 |
| 6.5 | Conclusões | 47 |
| 7 | Conclusões Finais e Perspectivas Futuras | 49 |
| | Referências Bibliográficas | 52 |
| A | Regras MiroSot | 56 |
| A.1 | O Campo e a Bola | 56 |
| A.2 | Os Jogadores | 57 |
| A.3 | Transmissão de Informações | 58 |
| A.4 | O Sistema de Visão | 58 |
| A.5 | O Jogo | 58 |
| B | Equações Paramétricas de Terceiro Grau | 61 |
| B.1 | Cálculo dos Parâmetros | 61 |
| B.2 | Casos Especiais | 62 |
| B.2.1 | Primeiro Caso | 62 |
| B.2.2 | Segundo Caso | 63 |
| B.2.3 | Terceiro Caso | 64 |
| B.3 | Determinação dos Coeficientes Livres | 64 |
| B.3.1 | Caso Geral | 67 |
| B.3.2 | Casos Especiais | 74 |

Lista de Figuras

| | | |
|-----|---|----|
| 2.1 | Estrutura do futebol de robôs na categoria <i>MiroSot-FIRA</i> | 6 |
| 2.2 | Campo de futebol com os minirobôs jogadores | 7 |
| 2.3 | Relação entre os módulos do projeto do futebol de robôs | 7 |
| 2.4 | Fluxo de dados do programa gerenciador do futebol de robôs | 8 |
| 3.1 | Exemplo da utilização do Diagrama de Voronöi para o planejamento de caminho em um ambiente fechado | 10 |
| 3.2 | Exemplo da utilização do Grafo de Visibilidade para o planejamento de caminho em um ambiente fechado | 11 |
| 3.3 | Mapeamento do campo de futebol de robôs em regiões de potencial atrativo e repulsivo (a) e o caminho percorrido pelo robô obtido a partir deste mapeamento (b) | 11 |
| 3.4 | Exemplo da utilização da Decomposição em Células Exatas para o planejamento de caminho em um ambiente fechado | 12 |
| 3.5 | Modelo do minirobô e seus parâmetros de configuração | 13 |
| 3.6 | Caminhos não-realizável (a) e realizável (b) por um veículo com restrição cinemática | 13 |
| 3.7 | Estratégia de controle para sistemas com restrições cinemáticas | 14 |
| 4.1 | Ausência dos dados x (—) e y (--) do minirobô | 18 |
| 4.2 | Valores medidos (—) e estimados (--) de x e y (a) e θ (b), com $n = 2$. Os respectivos erros absolutos são apresentados em (c) e (d) | 21 |
| 4.3 | Valores medidos (—) e estimados (--) de x , y e θ mostrados em (a) e (b), com $n = 5$. Os respectivos erros absolutos são apresentados em (c) e (d) | 22 |
| 4.4 | Valores medidos (—) e estimados (--) de x , y e θ , com $n = 10$, mostrados em (a) e (b), e os respectivos erros absolutos mostrados em (c) e (d) | 23 |
| 4.5 | Utilização dos valores estimados (--) nos instantes de ausência dos valores medidos (—) de x e y (a) e os respectivos erros absolutos (b) | 24 |
| 4.6 | Valores medidos (—) e estimados (--) de θ após normalização, com $n = 2$, (a) e erro absoluto encontrado (b) | 24 |
| 5.1 | Caminho composto por dois arcos de círculo de raio r e um segmento de reta | 26 |
| 5.2 | Caminho do minirobô aproximado por um arco de elipse | 27 |
| 5.3 | Parâmetros característicos de uma elipse | 27 |
| 5.4 | Conjunto de caminhos encontrados através da utilização de polinômios de segundo grau nas variáveis x e y de configuração | 30 |
| 5.5 | Caminhos encontrados através de polinômios de terceiro grau nas variáveis de configuração x e y | 31 |

| | | |
|------|--|----|
| 5.6 | Região final admissível para $x(\lambda)$ | 33 |
| 5.7 | Região admissível para $y(\lambda)$ | 33 |
| 5.8 | Caminhos encontrados através da utilização de polinômios de terceiro grau nas variáveis x e y , com os valores apropriados para os coeficientes livres | 34 |
| 5.9 | Trajetórias calculadas para serem desenvolvidas pelo minirobô | 36 |
| 6.1 | Diagrama das forças e torques que atuam sobre o minirobô | 39 |
| 6.2 | Análogo elétrico dos comportamentos linear e angular do robô | 40 |
| 6.3 | Análogo elétrico dos motores direito e esquerdo do minirobô | 41 |
| 6.4 | Estratégia do controle para aproximação da posição de referência final | 44 |
| 6.5 | Exemplos de estabilização do minirobô em uma configuração fixa | 45 |
| 6.6 | Comportamento em malha aberta do minirobô simulado, submetido a entradas iguais | 46 |
| 6.7 | Comportamento do minirobô quando são utilizados o planejamento de caminho, geração e controle de trajetória | 48 |
| A.1 | Medidas padrões do campo para a categoria MiraSot-FIRA | 60 |
| B.1 | Retângulo delimitado pelas posições inicial e final | 64 |
| B.2 | Região para a qual os valores de b e c implicam em raízes complexas para a equação B.7 | 65 |
| B.3 | Região na qual os valores dos coeficientes b e c implicam em raízes reais menores ou iguais a 0 para a equação B.7 | 66 |
| B.4 | Região de valores dos coeficientes b e c da equação B.7 que implicam em raízes reais maiores ou iguais a 1 | 66 |
| B.5 | Região para que a equação B.7 tenha uma raiz real menor ou igual a 0, e outra maior ou igual a 1 | 67 |
| B.6 | Região para que a equação B.7 não tenha valores máximos ou mínimos entre $0 < \lambda < 1$ | 67 |
| B.7 | Região delimitada pelas condições B.10 | 68 |
| B.8 | Região delimitada pelas condições B.11 | 68 |
| B.9 | Região delimitada pelas condições B.12 | 68 |
| B.10 | Região delimitada pelas condições B.13 | 69 |
| B.11 | Região delimitada pelas condições B.14 | 69 |
| B.12 | Região final dos coeficientes livres do caso geral para satisfazer o critério de aprimoramento na direção x | 70 |
| B.13 | Região final admissível para $y(\lambda)$ | 70 |
| B.14 | Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f > 0$ | 71 |
| B.15 | Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f < 0$ | 72 |
| B.16 | Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f > 0$ | 72 |
| B.17 | Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f < 0$ | 73 |
| B.18 | Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f = 0$ | 73 |

| | |
|--|----|
| B.19 Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f = 0$ | 73 |
| B.20 Região admissível para não haver máximos e mínimos na direção y quando $d_i = 0$ e $d_f > 0$ | 74 |
| B.21 Região admissível para não haver máximos e mínimos na direção y quando $d_i = 0$ e $d_f < 0$ | 74 |
| B.22 Região final admissível para o caso onde θ_i e θ_f são iguais a $\pm\pi/2$ | 75 |
| B.23 Região final admissível na direção x . Caso onde somente $\theta_i = \pm\pi/2$ | 75 |
| B.24 Região final admissível na direção y . Caso onde somente $\theta_i = \pm\pi/2$ | 75 |
| B.25 Região final admissível na direção x . Caso onde somente $\theta_f = \pm\pi/2$ | 76 |
| B.26 Região final admissível na direção y . Caso onde somente $\theta_i = \pm\pi/2$ | 76 |

Resumo

Um dos maiores problemas em robótica móvel diz respeito à sua navegação. Conceitualmente, o ato de navegar em robótica consiste em guiar um robô em um espaço de trabalho durante um determinado intervalo de tempo, por um caminho que possa ser percorrido e que leve o robô de uma posição e orientação iniciais para uma posição e orientação finais. Esta é a principal tarefa que um robô móvel deve executar. Ela implica em subproblemas que são a localização do robô no espaço de trabalho, o planejamento de um caminho admissível, a geração de uma trajetória e, por fim, a sua execução. Além disso, robôs móveis autônomos com rodas possuem restrições cinemáticas, chamadas também de restrições não-holonômicas, que fazem com que o robô não possa se mover livremente em seu espaço de trabalho, limitando a quantidade de caminhos admissíveis entre duas posições distintas. Este trabalho aborda principalmente os subproblemas do planejamento de caminho e geração de trajetória aplicado a minirobôs móveis com rodas que atuam em um projeto de futebol de robôs. O maior desafio para a navegação destes veículos é determinar uma função contínua que respeite suas restrições cinemáticas e evolua no tempo segundo as restrições impostas pelo problema quanto à posição e orientação iniciais e finais e quanto à velocidade do movimento. Propõe-se uma estratégia de geração de caminho baseada em polinômios paramétricos de terceiro grau em x e y . A orientação θ do minirobô é obtida da relação entre y e x de modo que os caminhos gerados respeitem a restrição cinemática imposta. Para que a trajetória seja executada e os resultados experimentais validados é apresentada uma estratégia simples de controle que atua sobre as velocidades linear e angular desenvolvidas pelo robô móvel.

Abstract

The main task and one of the major mobile robotics problems is its navigation process. Conceptually, this process means drive the robot from an initial position and orientation to a goal position and orientation, along an admissible path respecting the temporal and velocity constraints. This task must be accomplished by some subtasks like robot localization in the workspace, admissible path planning, trajectory generation and motion control. Moreover, autonomous wheeled mobile robots have kinematics constraints, also called nonholonomic constraints, that impose the robot can not move everywhere freely in its workspace, reducing the number of feasible paths between two distinct positions. This work mainly approaches the path planning and trajectory generation problems applied to wheeled mobile robots acting on a robot soccer environment. The major difficulty in this process is to find a smooth function that respects the imposed robot kinematic constraints. This work proposes a path generation strategy based on parametric polynomials of third degree for the x and y axis. The θ orientation is derived from the y and x relations in such a way that the generated path respects the kinematic constraint. To execute the trajectory, this work also shows a simple control strategy acting on the robot linear and angular velocities.

Capítulo 1

Introdução

Os robôs são uma tecnologia utilizada para auxiliar ou substituir o homem em tarefas que podem tornar-se corriqueiras, como em uma linha de produção industrial, ou em ambientes que conferem risco à vida humana como, por exemplo, o fundo do mar, proximidades de incêndios, terrenos minados ou áreas com contaminação radioativa. De uma maneira abrangente, os robôs podem ser definidos como máquinas que, além de serem capazes de reproduzir tarefas associadas aos seres humanos e movimentos implícitos em sua construção, complementam a parte mecânica com dispositivos eletrônicos inteligentes de suporte, os quais lhes dão um certo grau de autonomia [Med98a]. Outros conceitos atribuídos aos robôs são:

- segundo Latombe [Lat91], um robô é um órgão mecânico versátil equipado com atuadores e sensores sob o controle de um sistema computacional;
- para Russel e Norvig [RN95], robô é simplesmente um agente artificial e ativo cujo ambiente é o mundo físico.

A robótica é uma área multidisciplinar que abrange várias subáreas como eletrônica, controle, mecânica, inteligência artificial, programação paralela, comunicação de dados entre outras. As primeiras pesquisas em robótica ocorreram na década de 1960, com a criação de robôs para automatização do trabalho em fábricas.

Os robôs são classificados quanto à sua forma e função [RN95] podendo assim serem separados em manipuladores e em robôs móveis. Os robôs manipuladores caracterizam-se por apresentar o corpo, ou a sua estrutura, composta por elos rígidos que se ligam através de juntas. Em uma das extremidades da cadeia de elos encontra-se a base fixa do robô e na outra, o seu efetuator (órgão terminal). Este órgão é uma ferramenta que permite a iteração do robô com o ambiente, como por exemplo uma garra para a movimentação de peças em uma linha de produção [Cra86]. Os robôs móveis podem ser usados para substituir o homem em tarefas que exigem o deslocamento físico em um espaço de trabalho, tais como vigilância de prédios, no transporte de produtos em indústrias ou em ambientes perigosos para o ser humano. Estes tipos de robôs possuem variações como humanóides (antropomorfismo), robôs com pernas (*legged robots*) e com rodas (*wheeled robots*).

Robôs móveis com rodas são mais simples de serem construídos, mais eficientes e fáceis de controlar. As rodas permitem uma maior praticidade de locomoção em terrenos relativamente planos e dão um maior suporte estático ao robô [RN95]. Uma característica importante nestes robôs é o fato deles apresentarem restrições de movimento. Estas restrições caracterizam-se por não permitir que os robôs se desloquem livremente no seu

espaço de trabalho. Isto ocorre porque o controle de movimento é realizado por duas rodas, porém os robôs possuem três graus de liberdade que são a posição (x, y) e o ângulo de orientação θ . Restrições desta natureza são chamadas de não-holonômicas e um exemplo prático de um movimento que é influenciado por estas restrições é o estacionamento de um carro através do movimento de baliza.

A autonomia em robôs móveis é essencial em ambientes remotos, tais como planetas distantes, onde o tempo de comunicação entre o robô e o seu operador não permite a este a tomada de decisão em tempo real. Robôs autônomos devem apresentar reações às mudanças no ambiente, comportamento inteligente, integração entre vários sensores, resolução de múltiplas tarefas, robustez, operar sem falhas, programabilidade, modularidade, flexibilidade, expansibilidade, adaptabilidade e raciocínio global [Med98b].

O desenvolvimento de sensores é importante para que características como autonomia possam ser confiáveis, implicando na realização satisfatória das tarefas pelos robôs. Os sensores podem ser colocados no próprio robô (sensores *on-board*) fazendo com que ele identifique objetos, ou marcos, para realizar uma localização relativa de sua própria posição, ou podem ser colocados de uma maneira que permitam uma varredura geral do ambiente (sensores *off-board*) habilitando uma localização global. Atualmente os sensores visuais, tais como câmeras de vídeo, são bastante utilizados em sistemas robóticos. As imagens obtidas com câmeras são processadas, onde os objetos encontrados são aproximados por modelos geométricos pré-definidos na memória do computador que controla o robô (modelagem do ambiente) e auxiliam na construção de mapas para que seja realizada a navegação.

Normalmente, a navegação é a principal tarefa a ser executada por um robô móvel. Ela pode ser definida como os procedimentos necessários para que um sistema autônomo possa sair de sua configuração inicial e alcançar uma configuração final por um caminho que possa ser percorrido, sendo a configuração composta pela posição e orientação do robô no espaço de trabalho. A navegação consiste nos passos de localização do robô no ambiente em que ele está, no planejamento de um caminho entre a sua configuração atual até uma configuração final e da execução do seu movimento pelo caminho encontrado. Esta tarefa pode ser mais, ou menos, complexa de acordo com o ambiente do robô, pois o espaço de trabalho pode ou não estar livre de obstáculos, e caso existam, estes podem estar estáticos ou móveis.

A presença de múltiplos objetos movendo-se em um único espaço caracteriza um ambiente dinâmico. Ambientes assim apresentam desafios para pesquisadores pois englobam desde estudos em sensoriamento do ambiente até recursos de inteligência artificial. Estes ambientes também podem ser constituídos pela presença de vários robôs que trabalham de forma cooperativa. Esta é uma área emergente de estudos em robótica móvel e diz respeito à coordenação de tarefas entre diversos robôs com o objetivo de alcançarem uma meta comum. Um exemplo de um ambiente dinâmico bastante usado para estudos de cooperação entre múltiplos robôs, entre outros temas, é o do futebol de robôs onde uma equipe tem que cooperar entre si e de maneira conflitante com uma outra equipe adversária para alcançar os objetivos determinados. Esta é uma das mais flexíveis plataformas para experimentação de técnicas de processamento de imagens em tempo real, gerenciamento da atuação de várias máquinas autônomas trabalhando em conjunto com um objetivo comum, comunicação entre robôs e mecânica fina. Os resultados das pesquisas nesta área são de migração imediata para os ambientes industrial e científico [TF98].

O grupo de robótica do Departamento de Engenharia de Computação e Automação (DCA) da Universidade Federal do Rio Grande do Norte (UFRN) desenvolveu duas equipes para futebol de robôs além de toda a estrutura principal para o jogo, como campo e iluminação, pois é do seu interesse pesquisar os assuntos relacionados por esta plataforma, tais como processamento de imagens, inteligência artificial e sistemas de controle.

Para que a frota dos minirobôs execute suas tarefas, deve haver um sistema de navegação que garanta o movimento correto de cada robô no campo de futebol. Normalmente, um sistema de navegação é composto por uma etapa que encontra trajetórias realizáveis pelos robôs e por uma etapa posterior, que realiza o controle de baixo nível fazendo com que o robô execute o movimento planejado. Os robôs montados são minirobôs móveis sendo cada um com duas rodas, que estão ligadas a motores distintos, e acionadas de forma independente. Devido às rodas, eles apresentam a restrição não-holonomica citada. Assim, nem toda trajetória encontrada para realizar um determinado movimento é realizável pelo minirobô. Isto traz o primeiro problema para o sistema de navegação aqui proposto, que é encontrar uma trajetória que respeite a restrição cinemática do robô. Outro problema existente é que o minirobô é um sistema multivariável, apresentando duas variáveis de entrada (velocidades das duas rodas) para controlar três variáveis de saída (x, y, θ), o que impede a utilização de técnicas de controle clássico baseados sistemas de uma entrada e uma saída.

Devido a estas dificuldades dentro do contexto do futebol de robôs, este trabalho propõe um esquema de navegação que possa aplicado para uma frota de minirobôs móveis, satisfazendo às restrições impostas pelas características do sistema.

1.1 Objetivos

Este trabalho visa desenvolver algoritmos e *softwares* que permitam a navegação de uma frota de minirobôs, sob restrição não-holonomica, garantindo a execução da trajetória planejada pelo navegador.

O sistema de navegação possui uma dependência de outros sistemas que compõem o projeto do futebol de robôs, como será detalhado nos capítulos seguintes. O módulo de navegação possui um estimador para que as configurações dos robôs sejam calculadas entre intervalos de amostragem do sistema. Utilizando estes e outros dados fornecidos por outros módulos que compõem o projeto do futebol de robôs, pretende-se realizar um planejamento de trajetória que leve em conta a restrição não-holonomica dos robôs e que una a configuração atual do robô à configuração final através de um movimento contínuo. Assim, especificando os objetivos do trabalho, tem-se:

- desenvolver um sistema estimador para ser aplicado nos robôs do próprio time e nos adversários para garantir que eles sejam localizados no campo de futebol entre os instantes de amostragem do sistema;
- implementar um planejador de caminhos e um gerador de trajetórias que calcule as configurações de referência dos minirobôs a cada passo de amostragem do sistema, levando em conta a restrição cinemática dos robôs, e que ligue uma configuração inicial a uma configuração final através de um movimento contínuo;
- elaborar uma estratégia simples de controle para os minirobôs de forma que eles possam executar o movimento planejado, validando assim os métodos desenvolvidos para o planejamento de caminho e geração de trajetória.

1.2 Estrutura da Dissertação

Esta introdução visou fornecer os conceitos iniciais de robótica e para contextualizar o projeto do futebol de robôs como tema de pesquisa. No próximo capítulo, este projeto será detalhado de forma mais clara. Nele serão abordadas as características gerais do futebol de robôs, as características da categoria na qual o projeto da UFRN se inseriu e, por fim, as características do próprio projeto da UFRN apresentando três módulos de importância, que são a Visão, a Estratégia e a Navegação.

A partir do capítulo 3 será mostrado o tema desta dissertação que é a navegação. Pelo fato dela ser constituída das etapas de localização do robô, planejamento de caminho, geração e execução de trajetória, o capítulo enfatiza principalmente estes três últimos assuntos através de uma revisão bibliográfica do estado da arte. O capítulo 4 mostra o que foi desenvolvido para o auxílio da localização dos minirobôs em campo. Será apresentado o método de estimação e a técnica simples de filtragem das informações das configurações dos robôs jogadores. O capítulo 5 é dedicado ao estudo do planejamento de caminho e da geração de trajetória, apresentando também o que foi implementado. O capítulo 6 apresenta a técnica de controle utilizada para que os robôs executassem a trajetória encontrada. Serão apresentados os resultados obtidos com os programas desenvolvidos. O capítulo 7 mostra algumas conclusões para a técnica de navegação proposta e apresenta o que poderá ser realizado em futuros trabalhos para o projeto do futebol de robôs do DCA-UFRN. Por fim, os apêndices descrevem as regras do futebol para a categoria escolhida para este projeto e o desenvolvimento das ferramentas matemáticas utilizadas no planejamento de caminho.

Capítulo 2

Futebol de Robôs

Como foi mencionado no capítulo 1, o futebol de robôs é um laboratório multidisciplinar para pesquisas de robótica e áreas afins. Esta atividade possui regras de forma a uniformizar as equipes que queiram disputar algum campeonato. Neste capítulo serão apresentados as características do futebol de robôs, os aspectos gerais de uma das organizações que promovem os campeonatos mundiais e os aspectos gerais do projeto de futebol de robôs desenvolvido pelo grupo de pesquisa em robótica do DCA-UFRN.

2.1 Características Gerais

Assim como uma partida de futebol realizada por humanos, uma partida de futebol de robôs consiste na presença de duas equipes adversárias de robôs autônomos jogando entre si em um campo de futebol apropriado. Os jogadores são pequenos robôs que, dependendo da categoria em que estão, podem ter diferentes formas de locomoção. Nesta atividade é fundamental a autonomia dos robôs no campo. Todas as decisões sobre defesas, ataques e passes são realizadas por uma ou mais unidades de processamento. Operadores humanos não interferem na partida de futebol.

Existem diferentes tipos de equipes, campos e regras. Há equipes formadas por robôs humanóides, por cães robôs, por robôs com rodas ou até mesmo com agentes inteligentes em uma partida de futebol realizada apenas por *software*. O número de jogadores por equipe varia entre 3 e 11 robôs, de acordo com a categoria. Os robôs podem ter sensoria-mento embarcado, tal como *encoders*, sonares, câmeras de vídeo, ou a equipe pode dispor de uma câmera localizada acima do campo para fornecer uma imagem global do ambiente. O campo de futebol é feito de madeira com dimensões definidas pela organização. A cor do campo também muda de acordo com o sensor empregado. Caso utilize-se uma câmera para visão global, uma cor escura e fosca torna-se mais adequada.

Algumas equipes de futebol possuem robôs com unidades de processamento embarcadas. Outras optam por robôs mais simples controlados à distância através de um computador. Estas unidades de processamento são responsáveis pela interpretação dos dados dos sensores e pela tarefa da navegação. Caso a equipe utilize um computador (*host*) para o controle dos robôs, as informações são enviadas para os jogadores através de sinais de rádio-freqüência.

Atualmente há duas organizações que promovem o futebol de robôs internacionalmente. São elas a *RoboCup* e a *Federation of International Robot-soccer Association* (FIRA). Suas regras são basicamente as mesmas com relação ao tempo de duração da

partida, marcação de faltas, pênaltis e gols, mas diferenciam-se com relação ao número de robôs por equipe, dimensões e cores dos robôs e do campo. A equipe de futebol de robôs do DCA-UFRN optou por seguir o modelo definido pela FIRA devido ao baixo custo para implementar os robôs e a estrutura de jogo.

2.1.1 O Futebol da FIRA

A FIRA teve a sua origem na Coréia do Sul a partir do ano de 1995 no centro de pesquisa KAIST (*Korea Advanced Institute of Science and Technology*). Nesta data, o professor Jong-Hwan Kim fez um convite à comunidade científica em robótica para que fossem elaboradas regras básicas para uma área comum de pesquisa utilizando uma partida de futebol. Esta atividade ficou conhecida, posteriormente, como futebol de robôs. Sabia-se que com esta linha de pesquisa seria possível testar técnicas de processamento de sinais, controle, inteligência artificial e robótica [KVV98]. A FIRA organizou-se em categorias que diferenciam-se pelo número e dimensões dos robôs jogadores. Dentre estas categorias destaca-se a *Micro-Robot Soccer Tournament (MiroSot)* que utiliza minirobôs de dimensões igual ou inferior a um cubo de $7,5\text{cm}$ de lado e apenas três jogadores em cada equipe. Esta foi a categoria adotada pelo grupo do DCA-UFRN devido ao baixo custo de implementação.

Por serem de tamanho reduzido, os robôs não possuem uma unidade de processamento *on-board*. Esta função é destinada a um computador que envia os comandos para os minirobôs através de sinais de rádio. O computador é conectado a um transmissor e cada robô possui uma unidade receptora. O sensoriamento é feito por uma câmera de vídeo que é colocada a uma altura não inferior a 2m e que fornece uma imagem global para o computador, que então processa as imagens, retira as informações necessárias, toma decisões sobre os estados dos robôs e envia comandos para que eles executem tarefas especificadas pelo programa de controle. A estrutura física geral do futebol de robôs na categoria *MiroSot* é apresentada na figura 2.1 e algumas regras básicas do jogo são apresentadas no apêndice A.

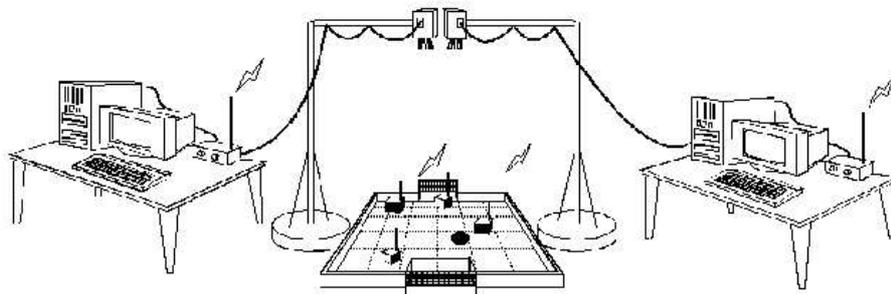


Figura 2.1: Estrutura do futebol de robôs na categoria *MiroSot-FIRA*

No Brasil o futebol de robôs tem-se destacado através das equipes dos centros de pesquisa do CTI, UFMG, UFRJ, UFRGS, UnB, PUC-RS e UFSC. Na região Nordeste destaca-se a equipe da UFRN, pioneira em usar o futebol como linha de pesquisa. Na próxima seção serão apresentadas as estruturas física e de *software* da equipe da UFRN.

2.2 O Futebol de Robôs do DCA-UFRN

A estrutura física do projeto do DCA-UFRN é composta por pelos mesmos elementos que compõem um equipe de futebol de robôs da categoria FIRA - *MiroSot*, ou seja, possui campo, minirobôs e computadores.

O campo de madeira é pintado de preto fosco para impedir a reflexão das luzes e possui textura similar a uma mesa de tênis-de-mesa. Ele possui uma parede de 5cm de altura na cor branca para impedir a saída da bola. Os seis minirobôs móveis são construídos em acrílico, com duas rodas de plástico com pneus de borracha. Para diferenciar uma equipe de outra utilizam-se etiquetas coloridas, onde as cores azul e amarela diferenciam as equipes e outras cores diferenciam os jogadores. A velocidade de cada roda é fornecida por um motor de corrente contínua sem redução de torque, acionados por pontes H e sinal PWM fornecido pelo microcontrolador. Os transmissores e os receptores trabalham com modulação FM e a alimentação é fornecida por baterias de 9V. Os dois computadores são ambos *Pentium III* e o programa de controle dos robôs foi escrito em linguagem C. Há também dois conjuntos transmissor-receptor que fazem a interface entre os computadores e os robôs e uma armação metálica para suportar a iluminação do campo e as câmeras de vídeo. A figura 2.2 mostra as duas equipes de minirobôs no campo de futebol.



Figura 2.2: Campo de futebol com os minirobôs jogadores

O projeto também consta de um programa para gerenciar a partida de futebol. Ele é dividido nos módulos de visão, estratégia e controle que executam tarefas próprias e de maneira síncrona. A figura 2.3 detalha o funcionamento e interligação entre eles.

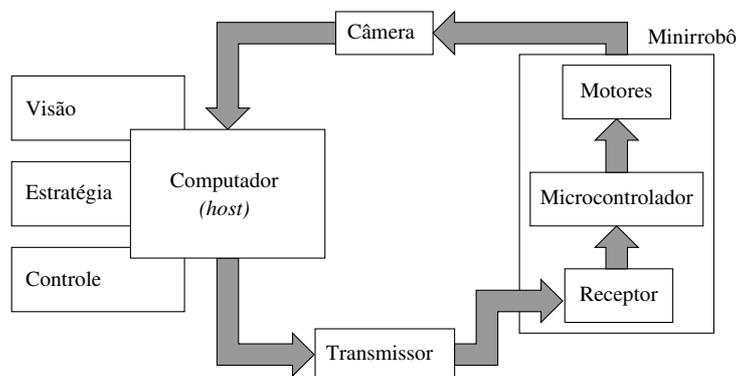


Figura 2.3: Relação entre os módulos do projeto do futebol de robôs

O sistema de visão além de compreender uma câmera de vídeo e um digitalizador, possui algoritmos para processamento de imagens. Elas são captadas pela câmera de vídeo e enviadas para o computador, onde são processadas pelo algoritmo de tratamento de imagens para serem extraídas as configurações dos robôs do próprio time, dos adversários e da bola. O desenvolvimento deste módulo foi realizado por Aires [Air01] no âmbito de outro trabalho de mestrado. O módulo de estratégia recebe os dados de configuração encontrados pelo sistema de visão e os interpreta para que sejam decididos os estados da equipe e de cada robô individualmente. A estratégia também informa aos outros módulos as configurações que os robôs têm que alcançar. Os estados são as ações que os robôs executam, tal como atacar, defender, recuar, passar a bola, entre outros. Este módulo também está sendo desenvolvido em outro trabalho de mestrado. O módulo de controle é responsável pela execução das tarefas planejadas pelo módulo de estratégia. Ele pode ser subdividido em estimação, planejamento de caminho, geração de trajetória e controle do robô, cujas descrições serão apresentadas nos próximos capítulos. O módulo de transmissão é executado de maneira assíncrona com os demais. Os sinais de controle são lidos e enviados via rádio aos robôs a cada $5ms$. Estes sinais são então recebidos e a eletrônica embarcada dos robôs converte-os em torque nos motores para gerar os movimentos no campo.

Percebe-se que a comunicação entre os módulos é importante pois alguns agem de maneira síncrona e outros, como a transmissão de dados, realizam-se assincronamente em tempo real. A figura 2.4 mostra o fluxo de dados do projeto do futebol de robôs.

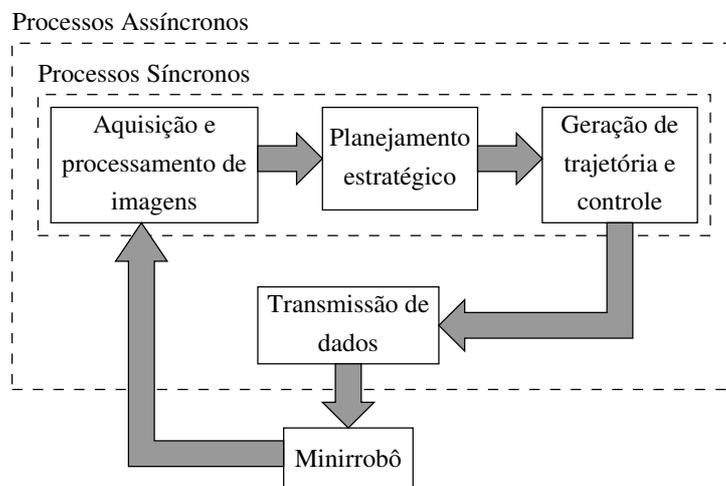


Figura 2.4: Fluxo de dados do programa gerenciador do futebol de robôs

2.3 Conclusões

Este capítulo apresentou as principais características do futebol de robôs e as regras para a categoria *MiroSot* da FIRA. Mostrou também a estrutura do projeto do futebol da UFRN e como os diversos módulos do projeto funcionam e se interligam. No próximo capítulo serão apresentados o estado da arte em navegação de robôs móveis e as técnicas adotadas para o projeto do futebol de robôs da UFRN.

Capítulo 3

O Problema da Navegação de Robôs Móveis com Rodas

Como visto no capítulo 1, os robôs móveis têm diversos tipos de atuadores para a sua locomoção, implicando assim em formas características como, por exemplo, robôs humanóides ou robôs semelhantes a carros. Neste caso, a primeira tarefa a ser cumprida é permitir que eles saiam de uma configuração inicial e alcancem uma configuração final desejada, em um tempo finito especificado. Isto é o chamado *problema da navegação*. Neste capítulo será abordado o estado da arte em navegação de robôs móveis, enfatizando o que foi realizado em termos de planejamento de caminho e controle de trajetória.

3.1 Navegação de Robôs

De um modo geral, a navegação é o processo de, a partir de uma posição e orientação atuais, alcançar uma posição e orientação finais em um determinado intervalo de tempo percorrendo, de forma contínua, posições e orientações intermediárias que possibilite o desvio de obstáculos, quando estes existirem. Para robôs móveis, a navegação é a tarefa básica e eles devem valer-se de seus recursos computacionais, sensores e atuadores para realizá-la adequadamente.

A navegação pode ser classificada como global, quando o ambiente de trabalho do robô é totalmente conhecido, ou local, quando o ambiente é parcialmente conhecido ou totalmente desconhecido. Ela é composta por etapas que, segundo Chatila [Cha95], são a percepção e modelagem do ambiente, a localização, o planejamento e decisão de movimento, e, por fim, a execução do movimento.

A percepção e modelagem do ambiente dizem respeito ao uso de sensores e da construção de modelos do espaço de trabalho do robô, para que este possa se localizar e localizar possíveis obstáculos. Todo o processo de navegação inicia-se com a utilização adequada de sensores.

Anousaki e Kyriakopoulos [AK99] e Oriolo *et al* [OUV98] utilizaram ultra-sons em seus sistemas de navegação. Sensores óticos foram empregados por Howard e Kitchen [HK99] e por Borges [Bor98]. Araújo e Almeida [AA99] projetaram um sistema de navegação que possui um aprendizado com redes neurais e lógica nebulosa a partir das informações vindas de diversos sensores. Veelaert e Peremans [VP99] fizeram a construção de mapas para realizarem a navegação de robôs também através de sensores. Foux *et al* [FHB93] também

realizaram a construção de mapas a partir da varredura do ambiente de trabalho do robô móvel por sensores *lasers*. Fujimori *et al* [FNG97] utilizaram um sistema de navegação adaptativa, onde o ângulo de orientação de referência é selecionado adaptativamente de acordo com a medição de três sensores de distância presentes no corpo do robô.

Pelo fato das tarefas de planejamento e execução do movimento serem abordados neste trabalho, as próximas seções apresentam de maneira mais detalhada cada um destes aspectos, assim como o estado da arte dos mesmos.

3.2 Planejamento e Geração do Movimento

O planejamento de caminhos e geração de trajetórias têm a finalidade de encontrar um caminho admissível entre duas configurações distintas, que possa ser percorrida em um determinado intervalo de tempo. O caminho é a união física entre dois pontos através de um conjunto de pontos intermediários. A trajetória consiste em obter perfis de velocidade para o robô para que ele possa percorrer o caminho encontrado.

Existem várias técnicas para o planejamento de caminho e a maioria delas foi compilada por Latombe [Lat91]. Ele apresentou os métodos de planejamento mais comuns, que empregam um modelo bidimensional e aproximam o robô e os obstáculos por polígonos. Dentre os principais métodos destacam-se o *Mapa de Rotas*, o *Campo de Potencial* e a *Decomposição em Células*.

O *Mapa de Rotas* consiste em obter um grafo não direcionado que represente a conectividade do espaço livre do ambiente e a partir dele encontrar um caminho adequado entre as posições inicial e final do robô. Algumas ferramentas como o Diagrama de Voronöi e Grafo de Visibilidade, exemplificados respectivamente nas figuras 3.1 e 3.2, são usados para obter este grafo.

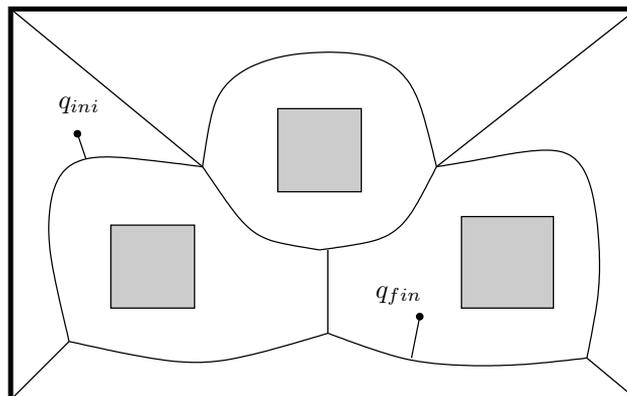


Figura 3.1: Exemplo da utilização do Diagrama de Voronöi para o planejamento de caminho em um ambiente fechado

Em seu trabalho, Choset e Burdick [CB95a, CB95b] utilizaram o Diagrama de Voronöi para obter a conectividade do espaço livre e para a construção de mapas usando apenas dados de sensores de distância. León e Sossa [LS98] utilizaram uma variação do método de esqueletização das imagens do espaço de trabalho do robô para obter um caminho livre entre os obstáculos.

A abordagem de campo de potencial mapeia o espaço de trabalho do robô em regiões de potencial artificial atrativo e repulsivo. Desta forma o robô é considerado como um ponto

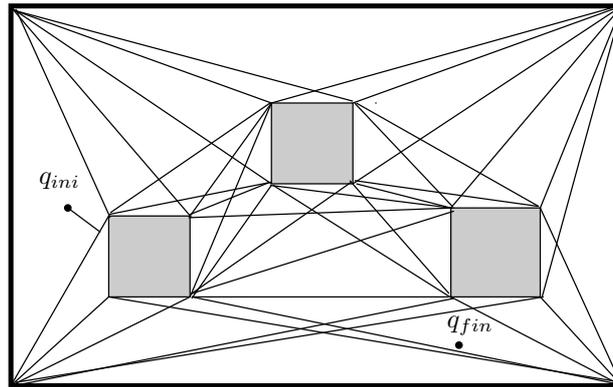


Figura 3.2: Exemplo da utilização do Grafo de Visibilidade para o planejamento de caminho em um ambiente fechado

móvel que pode ser guiado através deste potencial, de sua posição atual até a posição final desejada, que é a região de mínimo global. A figura 3.3 apresenta um resultado obtido por Aminadab e Alsina [SA00] em um estudo de caso do campo de potencial em ambientes de futebol de robôs.

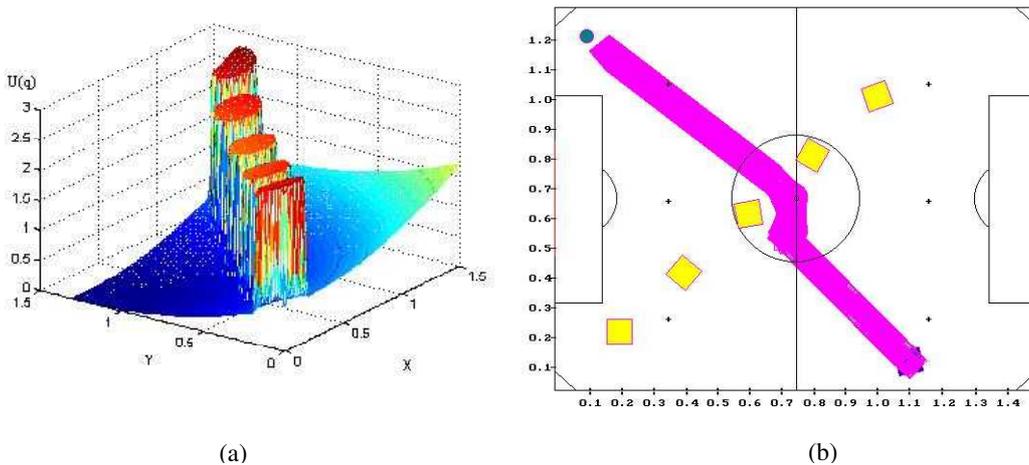


Figura 3.3: Mapeamento do campo de futebol de robôs em regiões de potencial atrativo e repulsivo (a) e o caminho percorrido pelo robô obtido a partir deste mapeamento (b)

Chuang e Ahuja [CA98] usaram o campo de potencial, para o planejamento de caminhos, baseado na função de potencial Newtoniano. Outros trabalhos que utilizaram esta abordagem foram desenvolvidos por Hwang e Ahuja [HA92], Rimon e Koditscheck [RK92] e Kassim e Kumar [KK99].

A Decomposição do espaço livre do robô em células consiste em separar este espaço em regiões convexas simples chamadas de células. Um grafo não-direcionado é então criado, baseando-se na adjacência entre estas células, e pesquisado de modo a obter o caminho entre as posições inicial e final. Existem dois métodos que utilizam este princípio: decomposição exata, cuja união das células é exatamente o espaço livre do robô, como exemplificado na figura 3.4, e a decomposição aproximada, cuja união das células está contida no espaço livre do robô. Um trabalho que utiliza este método é o apresentado por de Chen *et al* [CZ97].

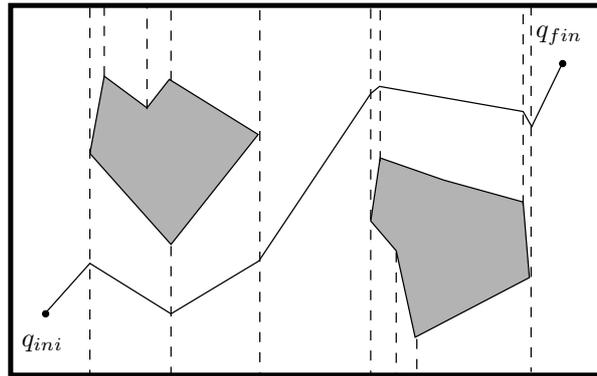


Figura 3.4: Exemplo da utilização da Decomposição em Células Exatas para o planejamento de caminho em um ambiente fechado

3.2.1 Restrições Não-Holonômicas

A maioria das técnicas de planejamento de caminho compiladas por Latombe [Lat91] leva em consideração apenas as restrições dinâmicas do ambiente de trabalho do robô, ou seja, a presença de obstáculos. O caminho gerado é livre de colisões e consiste na concatenação de segmentos de reta, o que lhe fornece a denominação de caminho geométrico.

Robôs móveis com rodas possuem restrições cinemáticas, chamadas de restrições não-holonômicas, que surgem normalmente do contato entre um objeto rolante e um objeto rígido, como as rodas do robô e o piso do ambiente de trabalho. As restrições não-holonômicas impedem que o robô execute alguns tipos de movimento. Por exemplo: robôs com rodas podem se mover para frente e para trás. Eles também podem realizar curvas, porém não podem movimentar-se de lado. Assim, estas restrições normalmente impedem que o robô percorra um caminho geométrico pois reduzem o conjunto de velocidades que o robô pode assumir. Matematicamente, as restrições não-holonômicas são caracterizadas por uma equação não integrável que envolve as derivadas no tempo dos parâmetros de configuração.

Considerando o modelo minirobô de duas rodas mostrado na figura 3.5, as variáveis de configuração são a sua posição e orientação (x, y, θ) . Como a restrição não-holonômica diz que o robô pode mover-se somente para onde a sua orientação aponta, da figura 3.5 pode-se tirar a seguinte relação:

$$\tan \theta(t) = \frac{\sin \theta(t)}{\cos \theta(t)} = \frac{\dot{x}(t)}{\dot{y}(t)}$$

obtendo assim a equação que descreve a relação não holonômica:

$$-\dot{x}(t) \sin \theta(t) + \dot{y}(t) \cos \theta(t) = 0$$

Um exemplo clássico de movimento que sofre restrições não-holonômicas é o estacionamento de um carro através da baliza, como exemplificado na figura 3.6.

Um dos métodos mais comuns de planejamento de movimento para robôs sujeitos a esta restrição cinemática é dividir o problema do planejamento do caminho em dois subproblemas distintos:

1. construir um caminho geométrico, ignorando as restrições não-holonômicas, utilizando algum método abordado por Latombe [Lat91];

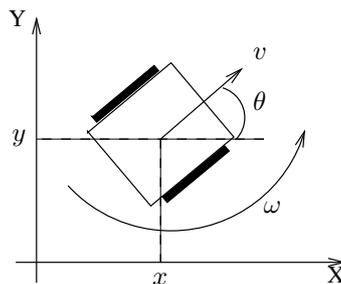


Figura 3.5: Modelo do minirobô e seus parâmetros de configuração

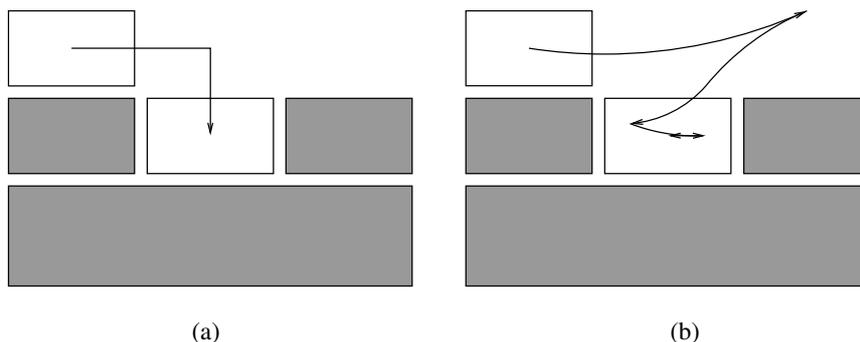


Figura 3.6: Caminhos não-realizável (a) e realizável (b) por um veículo com restrição cinemática

2. transformar o caminho geométrico encontrado em um caminho que possa ser percorrido pelo robô e topologicamente equivalente ao caminho geométrico.

Há várias propostas para a transformação deste caminho geométrico em um caminho que atenda às restrições cinemáticas do robô. O método clássico empregado para robôs móveis semelhantes a carros foi proposto por Reeds e Sheep [RS90]. Eles mostraram que entre duas configurações distintas existe um caminho composto por arcos de círculo de raio mínimo e segmentos de reta que é o mais curto dentre um conjunto de caminhos com as mesmas características, entretanto as trajetórias realizadas nestes caminhos implicam que o robô deve parar o seu movimento e realizar mudanças no sentido da sua velocidade. A desvantagem deste método é que ele pode incluir um caminho com muitos movimentos reversos [Lat91].

Muitos dos trabalhos sobre planejamento de trajetórias para robôs móveis não-holonômicos foram desenvolvidos por Laumond *et al* [LJTM94, Lau98]. Ele apresentou um método de planejamento para um robô móvel com rodas, com raio de curvatura mínimo, que foi aplicado para uma família de robôs. Ferbach [Fer98] apresentou um método para planejamento de movimento para robôs não-holonômicos com *trailers*. Em seu trabalho, primeiramente encontra-se um caminho que satisfaça somente às restrições dinâmicas para, em seguida, transformá-lo iterativamente em um caminho realizável, na medida em que as restrições cinemáticas vão sendo introduzidas. Divilbiss e Wen [DW97] também apresentaram um algoritmo para encontrar um caminho cinematicamente realizável na presença de obstáculos. Assim como outros trabalhos na área, Divilbiss e Wen também dividem o problema em planejamento de caminho geométrico e adaptação do caminho de acordo com as restrições cinemáticas.

3.3 Controle do Movimento

De uma maneira geral, técnicas clássicas de controle não são suficientes para garantir que um robô móvel cumpra a sua tarefa. Isso ocorre porque o robô com rodas é um sistema multivariável, com as velocidades linear e angular para controlar a posição x, y e orientação θ , não-linear e com acoplamento nas variáveis de saída. Além disso, a estabilidade é garantida apenas quando o robô trabalha linearizado em um ponto de operação [Per00].

Para garantir a controlabilidade do robô, divide-se o sistema de controle em dois, como apresentado na figura 3.7. Um é feito em malha aberta e é um gerador de referências para o controlador propriamente dito, em malha fechada. A princípio, o controle em malha aberta de um robô garante a geração exata do caminho desejado, mas na prática não pode ser utilizada isoladamente pois baseia-se em um conhecimento completo dos modelos do robô e dos atuadores, o que não pode ser garantido. Os erros não são corrigidos e tornam-se acumulativos [Med98a].

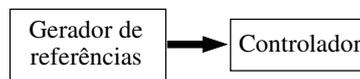


Figura 3.7: Estratégia de controle para sistemas com restrições cinemáticas

Alguns trabalhos de controle de trajetória de robôs móveis podem ser citados como o de Lages e Hemerly [LH00] que utilizaram técnicas de controle adaptativo para robôs móveis, onde o controlador baseia-se na informação de que o modelo do robô é formado por dois sistemas dinâmicos em cascata. Eles propuseram um controlador não-linear para a cinemática do robô e uma malha interna de controle de velocidade, utilizando controle linearizante adaptativo. Yang [YK98] propôs uma lei de controle por modos deslizantes para o problema do robô rastrear uma trajetória. Souza e Hemerly [SJH00] realizaram o controle de robôs móveis através de uma rede neural artificial para ajuste dos parâmetros do modelo de robô. Pereira [Per00] projetou um controlador para robôs móveis externamente controlados e observados por visão. Ele desacoplou as variáveis de controle e com isso conseguiu projetar controladores clássicos para duas malhas distintas. O modelo do robô foi obtido através das equações cinemáticas e dinâmicas e os seus parâmetros por métodos de identificação.

3.4 Técnicas Utilizadas no Futebol de Robôs

Para o futebol de robôs, algumas observações têm que ser feitas. Primeiro, ele consiste em um ambiente dinâmico e não-controlável, o que pode ser comprovado pelos vários robôs móveis movimentando-se em campo e pelo desconhecimento das trajetórias que os adversários executarão. Pelo fato de ser dinâmico, as técnicas padrões de planejamento de caminho não se aplicam, pois o cenário sofre alterações a todo instante de tempo. Outra restrição é que o planejamento do caminho, geração da trajetória e a geração de sinais de controle têm que ser feitos durante os intervalos de amostragem do sensor. Assim, estas tarefas não podem ser demasiadamente longas pois isso inviabilizaria a ação das outras tarefas. Outros fatores restritivos no futebol de robôs são o tempo de atraso existente entre o envio do sinal de controle e a resposta propriamente dita do robô, e situações não previstas como colisões com os robôs adversários. Todas estas peculiaridades fazem com

que o ambiente de futebol de robôs necessite de algoritmos eficientes para que se possam obter bons resultados.

Segundo Pereira [PCA00], o controle de robôs no futebol é difícil devido aos atrasos do processamento da imagem e do sistema transmissão. Para Costa [CPS⁺99, CP00] e a equipe de futebol Guaraná, é importante prever os estados futuros dos robôs e da bola no futebol devido ao ruído no sistema e ao *delay* da comunicação. A predição de parâmetros, tais como a posição dos obstáculos, é importante para o planejamento de trajetória dos robôs onde, entre outras coisas, pode-se evitar colisões futuras com obstáculos. Elnagar e Gupta [EG98] propuseram um algoritmo que prediz as configurações futuras dos obstáculos, que se movem livremente, baseados em um modelo autoregressivo com condicional de máxima verossimilhança para estimar os parâmetros do modelo. O seu algoritmo desenvolvido é capaz de prever a $(n + 1)$ -ésima configuração após as últimas n observações. Em seu trabalho, Pereira [PCA00] utilizou algoritmos de predição para encontrar os parâmetros do modelo matemático dos robôs e a configuração dos mesmos durante uma partida de futebol. Costa [CP00] usou uma função ajustada empiricamente para estimar a posição dos robôs. Quanto aos adversários e à bola, a estimação foi feita a partir da análise das imagens atual e anterior. O time *CMUnited* [HV98] utilizou um filtro de Kalman estendido para a predição das localizações futuras da bola.

Quando se trata de planejamento de caminhos, percebe-se que algoritmos simples são usados em virtude dos obstáculos serem móveis. Para a equipe *CMUnited* [HV98], o planejamento de caminho livre dos robôs usou um controle realimentado para calcular a sua posição em torno dos obstáculos. O algoritmo consiste em uma geração incremental de diferentes possíveis alvos intermediários. Este algoritmo não impede a colisão dos robôs com os adversários e paredes do campo. O algoritmo da equipe Guaraná [CPS⁺99, CP00] gera uma linha reta entre o robô e o alvo e verifica se existe algum robô adversário sobre ela. Caso exista, um alvo provisório é calculado 20cm distante do adversário, levando em consideração as paredes do campo.

Com relação ao controle de trajetórias, somente o trabalho de Pereira [Per00] mostrou-se detalhado. Ele modelou os robôs móveis através das suas equações cinemáticas e dinâmicas e, através de técnicas de identificação, ele calculou os valores dos parâmetros. Com isso, foi possível encontrar a função de transferência do robô e projetar dois controladores clássicos, um para a malha de controle de movimentos lineares e outro para movimentos angulares.

3.5 Propostas para a Equipe do DCA-UFRN

Pelos trabalhos apresentados na área de futebol de robôs vê-se que a utilização de algoritmos mais simples e menos custosos computacionalmente oferece resultados satisfatórios. Seguindo esta linha de raciocínio, a proposta da equipe do DCA-UFRN para o planejamento de caminhos e geração de trajetória é apresentado neste trabalho.

Inicialmente deve-se garantir que, a todo instante de amostragem, o sistema disponha das configurações dos minirobôs. Normalmente estas informações são fornecidas pelo sistema de visão mas devido a algum problema como iluminação irregular sobre o campo, o sistema de visão não consegue calcular a configuração dos minirobôs em alguns momentos esporádicos. Esta ausência de dados pode resultar em problemas para os outros módulos, como o controle por exemplo. Para resolver isto, utilizou-se o algoritmo dos Mínimos

Quadrados para estimar a configuração dos minirobôs. Estes dados são calculados a todo instante de amostragem mas são utilizados apenas quando o sistema de visão não fornece as informações necessárias. Este procedimento é similar ao utilizado por Elnagar e Gupta [EG98].

Uma vez tendo os valores das configurações dos robôs, passa-se a realizar o planejamento do caminho a cada passo de amostragem. Para esta tarefa interpola-se a atual configuração do robô com a configuração final desejada através de uma função paramétrica de terceiro grau em x e y . Este planejamento não necessita encontrar um caminho geométrico para, em seguida, transformá-lo em um caminho realizável, pois o seu cálculo é realizado sem a preocupação com colisões. Como o caminho é recalculado a todo instante, alguma técnica de desvio local pode facilmente impedir alguma colisão. Ao gerar um caminho através de polinômios paramétricos de terceiro grau, respeita-se a restrição não-holonômica pois ela é incluída no cálculo do caminho. Para a geração de trajetória, calcula-se o comprimento do caminho encontrado e através do Método da Bisseção determina-se o valor do parâmetro do polinômio que é correspondente à configuração desejada para o robô no próximo intervalo de amostragem.

Para a execução da trajetória, optou-se por trabalhar com variáveis de controle desacopladas. Foi implementado um controlador PID simples capaz de seguir uma referência variante no tempo, como a configuração calculada pelo gerador de trajetória. Semelhantemente à proposta de Pereira [Per00], o modelo do minirobô foi extraído das equações cinemáticas e dinâmicas, porém o modelo apresentado neste trabalho apresenta-se mais completo pois inclui a dinâmica dos atuadores utilizados nos robôs.

3.6 Conclusões

Este capítulo apresentou uma visão geral do estado da arte em navegação de robôs móveis autônomos. Foram mostrados as técnicas empregadas em outras equipes de futebol de robôs e o que será utilizado no projeto do DCA-UFRN. Os próximos capítulos estão dedicados ao trabalho realizado para o planejamento de movimento e controle dos robôs. No capítulo seguinte será apresentada a proposta do algoritmo de estimação utilizado para prever a configuração dos robôs no campo e garantir que todos os módulos do projeto tenham, a cada intervalo de amostragem, os dados necessários para realizarem suas tarefas.

Capítulo 4

Estimação da Configuração dos Minirroboês

Nos capítulos anteriores foram apresentados os principais conceitos de robótica e futebol de robôs, além do estado da arte em planejamento de caminhos e controle de trajetória para robôs móveis autônomos. Este capítulo apresenta o algoritmo de estimação utilizado para calcular a configuração do minirroboê e a razão da utilização desta técnica.

4.1 Introdução

A determinação da posição e orientação dos objetos presentes em um campo de futebol de robôs é uma tarefa associada ao módulo de visão do projeto do DCA-UFRN. Porém, em experiências com os minirroboês reais percebeu-se que em instantes aleatórios de tempo o sistema de visão não havia calculado a configuração dos robôs. Esta ausência de dados é possivelmente decorrente de problemas com a iluminação, que durante o movimento dos robôs no campo, dificulta a determinação da configuração dos robôs. Um exemplo deste problema pode ser visualizado na figura 4.1.

Dessa forma, uma etapa de estimação da configuração dos minirroboês foi importante devido ao fato de que ausência de dados implicam em um desempenho irregular das funções de planejamento de caminho, geração de trajetória e controle dos robôs. Além disso, uma predição da posição dos robôs é útil para elaboração de estratégias mais adequadas como, por exemplo, prever a melhor configuração para o robô de modo que ele efetue uma defesa ou ataque bem sucedido.

Existem diversos algoritmos de estimação de parâmetros, sendo cada um adequado a um tipo de problema. Há métodos que realizam a estimação de modo *off-line*, ou seja, após a aquisição dos dados necessários ao problema, e métodos que trabalham no modo *on-line*, onde a estimação é realizada na medida em que os dados são adquiridos. Dentre os mais comuns podem ser citados o algoritmo dos *Mínimos Quadrados*, o método da *Máxima Verossimilhança* e técnicas de estimação utilizando o *Filtro de Kalman*. A técnica empregada para realizar esta tarefa foi o algoritmo dos Mínimos Quadrados, pois ele mostrou-se simples e bastante abordado na literatura, como em Åström e Wittenmark [ÅW97] e em Aguirre [Agu00].

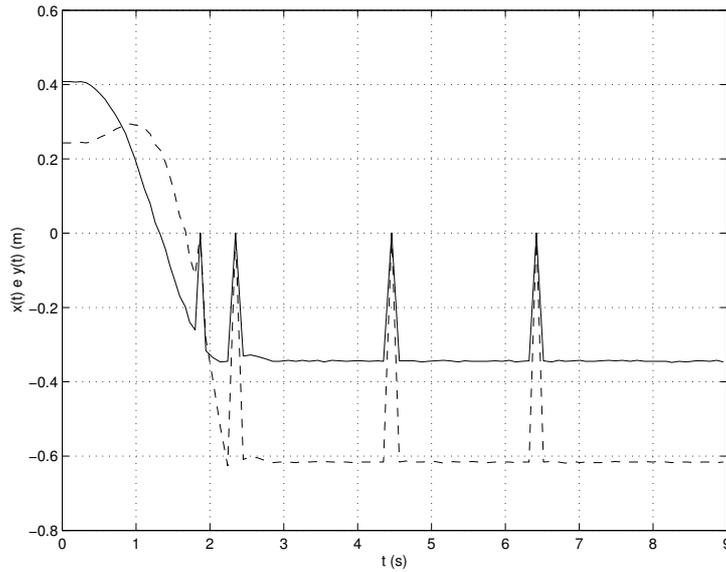


Figura 4.1: Ausência dos dados x (—) e y (---) do minirobô

4.2 Algoritmo dos Mínimos Quadrados

A idéia principal do algoritmo dos Mínimos Quadrados é que os parâmetros desconhecidos de uma determinada função $y = f(\mathbf{x})$ possam ser escolhidos de tal forma que a equação 4.1 possa ser minimizada.

$$J = \|\mathbf{y} - \hat{\mathbf{y}}\|_2 \quad (4.1)$$

O vetor \mathbf{y} é chamado de vetor de dados observados. Ele é composto por N elementos (ou N observações) na forma:

$$\mathbf{y} = [y_1 \quad y_2 \quad \dots \quad y_N]^T$$

cujos elementos são calculados através de:

$$y_i = f(\mathbf{x}_i)$$

onde i indica o i -ésimo termo, $f(\cdot)$ é uma função linear com parâmetros desconhecidos e \mathbf{x} (equação 4.2) é um vetor de entradas variantes no tempo com n termos.

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]^T \quad (4.2)$$

Sendo $f(\cdot)$ uma função linear definida em termos de parâmetros, define-se:

$$\boldsymbol{\theta} = [\theta_1 \quad \theta_2 \quad \dots \quad \theta_n]^T$$

onde θ_i , com $i = 1, \dots, n$, é o i -ésimo parâmetro desconhecido. Assim:

$$y_i = \theta_1 x_{1,i} + \theta_2 x_{2,i} + \dots + \theta_n x_{n,i}$$

Como são necessários N medições de y , recai-se em um sistema linear do tipo:

$$\mathbf{y} = \boldsymbol{\theta}^T X$$

onde $X \in \mathbb{R}^{n \times N}$ (equação 4.3) é a matriz que contém os N dados de entrada.

$$X = [\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_N] = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,N} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,N} \\ \vdots & \vdots & \cdots & \vdots \\ x_{n,1} & x_{n,2} & \vdots & x_{n,N} \end{bmatrix} \quad (4.3)$$

O vetor $\hat{\mathbf{y}}$ é o vetor de dados estimados. Da mesma forma que \mathbf{y} , ele possui N elementos e é definido como:

$$\hat{\mathbf{y}} = \hat{\boldsymbol{\theta}}^T X$$

sendo $\hat{\boldsymbol{\theta}}$ o vetor de parâmetros estimados. É este vetor que deve ser calculado a fim de se obter $\hat{\mathbf{y}}$ e minimizar a equação 4.1.

Há duas formas de calcular $\hat{\boldsymbol{\theta}}$: a estimação em batelada, que necessita das N observações de saída, e a estimação iterativa, onde o cálculo é realizado à medida que os dados vão sendo adquiridos. Para isto, necessita-se de um vetor de ganhos \mathbf{k} , de n elementos, e de uma matriz de covariância $P \in \mathbb{R}^{n \times n}$. Para calcular o vetor $\hat{\boldsymbol{\theta}}$ utilizam-se as seguintes equações:

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \hat{\boldsymbol{\theta}} + \mathbf{k}(y - \hat{\boldsymbol{\theta}}^T \mathbf{x}) \\ \mathbf{k} &= P\mathbf{x}(\lambda + \mathbf{x}^T P\mathbf{x})^{-1} \\ P &= \frac{1}{\lambda}(I - \mathbf{k}\mathbf{x}^T)P \end{aligned}$$

O termo λ é conhecido como fator de esquecimento. Ele indica quão rápido os valores passados são ignorados. Normalmente, o seu valor é menor que 1 [ÅW97].

4.3 Aplicação no Futebol de Robôs

As variáveis de configuração (x, y, θ) dos minirobôs são descritas por equações não-lineares que, por sua vez, são inadequadas para o método dos Mínimos Quadrados. Isto a princípio inviabilizaria a sua utilização, porém como o objetivo de uma etapa de predição da configuração dos minirobôs é garantir que os módulos de estratégia e controle tenham as informações necessárias apenas nas ausências dos dados fornecidos pelo sistema de visão, e não descrevê-los fielmente durante a partida de futebol, pode-se perfeitamente utilizar modelos lineares que descrevem a posição e orientação dos minirobôs apenas baseados em valores passados. Desta forma, nesta etapa dispensa-se o uso do modelo dinâmico completo do robô. Isto se torna mais simples pelo fato de que estas ausências ocorrem de maneira aleatória e esparsa. Assim, considerando a k -ésima iteração, as variáveis x , y e θ foram linearizadas como:

$$\begin{aligned} x_k &= \alpha_1^x x_{k-1} + \cdots + \alpha_n^x x_{k-n} + \beta_1^x u_{k-1}^e + \cdots + \beta_n^x u_{k-n}^e + \gamma_1^x u_{k-1}^d + \cdots + \gamma_n^x u_{k-n}^d \\ y_k &= \alpha_1^y y_{k-1} + \cdots + \alpha_n^y y_{k-n} + \beta_1^y u_{k-1}^e + \cdots + \beta_n^y u_{k-n}^e + \gamma_1^y u_{k-1}^d + \cdots + \gamma_n^y u_{k-n}^d \\ \theta_k &= \alpha_1^\theta \theta_{k-1} + \cdots + \alpha_n^\theta \theta_{k-n} + \beta_1^\theta u_{k-1}^e + \cdots + \beta_n^\theta u_{k-n}^e + \gamma_1^\theta u_{k-1}^d + \cdots + \gamma_n^\theta u_{k-n}^d \end{aligned}$$

onde α_j^i , β_j^i e γ_j^i , com $i = x, y, \theta$ e $j = 1, \dots, n$ são os parâmetros a serem calculados, n é a quantidade total de valores passados que são utilizados; x_{k-j} , y_{k-j} e θ_{k-j} são os valores

passados utilizados; u_{k-j}^e e u_{k-j}^d são as entradas normalizadas dos motores esquerdo e direito, respectivamente.

Um outro problema foi o valor do termo n . Como não se tinha idéia da ordem do sistema real, optou-se por encontrá-lo empiricamente, testando os valores 2, 5 e 10 e verificando qual apresentaria melhor comportamento dos valores estimados. Os valores iniciais dos parâmetros desconhecidos foram escolhidos de maneira aleatória e os elementos da diagonal principal da matriz de covariância P foram inicializados com valores iguais a 1.000.000, correspondendo a um erro alto entre os valores dos parâmetros iniciais e reais. O fator de esqucimento λ foi igual a 0,995.

Uma vantagem do método utilizado é que o cálculo dos valores estimados pode ser realizado iterativamente, evitando trabalhar com matrizes de dimensões grandes e tornando o programa mais rápido. Um outro motivo para utilizar o modo iterativo foi o fato de que os valores dos parâmetros eram válidos apenas para o passo de amostragem em que eles foram calculados.

As figuras seguintes mostram os testes realizados com um minirobô para a identificação da ordem n do sistema. Os motores do minirobô utilizado foram ambos submetidos a uma entrada degrau. O movimento foi realizado em malha aberta. Para estes testes inicialmente coletou-se um conjunto de dados de configuração que foram armazenados e utilizados *a posteriori* para a validação do método adotado.

A figura 4.2 apresenta o resultado da medição e da estimação nas variáveis x , y e θ , utilizando $n = 2$. Os valores medidos pelo sistema de visão são apresentados com a linha contínua, enquanto os valores estimados são apresentados com uma linha tracejada. Pode-se notar nos valores medidos a ausência das informações do sistema de visão próximas dos instantes 2s e 9s. Nesta figura também são apresentados os respectivos erros absolutos entre os valores medidos e os estimados.

A figura 4.3 mostra os resultados para x , y e θ com $n = 5$. Os erros absolutos encontrados entre os valores observados e os calculados também são apresentados.

Por fim, a figura 4.4 apresenta os resultados da estimação de x , y e θ , para $n = 10$, juntamente com os erros absolutos encontrados.

Pelos resultados apresentados percebe-se que ocorre um aumento do erro dos valores estimados das variáveis de configuração em relação às variáveis medidas pelo sistema de visão à medida que aumenta-se o valor de n . Isto se dá principalmente nos instantes iniciais da estimação e durante as ausências dos dados de configuração, que se propaga nas n amostras posteriores. Diante disto, verificou-se que um sistema com ordem 2 atenderia perfeitamente às exigências pois os erros são menores quando comparados com outros resultados e, em caso de falha na medição, este erro propaga-se somente por mais duas iterações. Assim, têm-se:

$$\begin{aligned} x_k &= \alpha_1^x x_{k-1} + \alpha_2^x x_{k-2} + \beta_1^x u_{k-1}^e + \beta_2^x u_{k-2}^e + \gamma_1^x u_{k-1}^d + \gamma_2^x u_{k-2}^d \\ y_k &= \alpha_1^y y_{k-1} + \alpha_2^y y_{k-2} + \beta_1^y u_{k-1}^e + \beta_2^y u_{k-2}^e + \gamma_1^y u_{k-1}^d + \gamma_2^y u_{k-2}^d \\ \theta_k &= \alpha_1^\theta \theta_{k-1} + \alpha_2^\theta \theta_{k-2} + \beta_1^\theta u_{k-1}^e + \beta_2^\theta u_{k-2}^e + \gamma_1^\theta u_{k-1}^d + \gamma_2^\theta u_{k-2}^d \end{aligned}$$

Uma vez obtidos, estes valores estimados são utilizados pelos outros módulos do projeto apenas nos instantes de falha de medição. Como resultado final tem-se que, durante todo o instante de uma partida de futebol, são asseguradas as informações sobre a configuração dos minirobôs, evitando assim erros no planejamento de caminho ou no controle

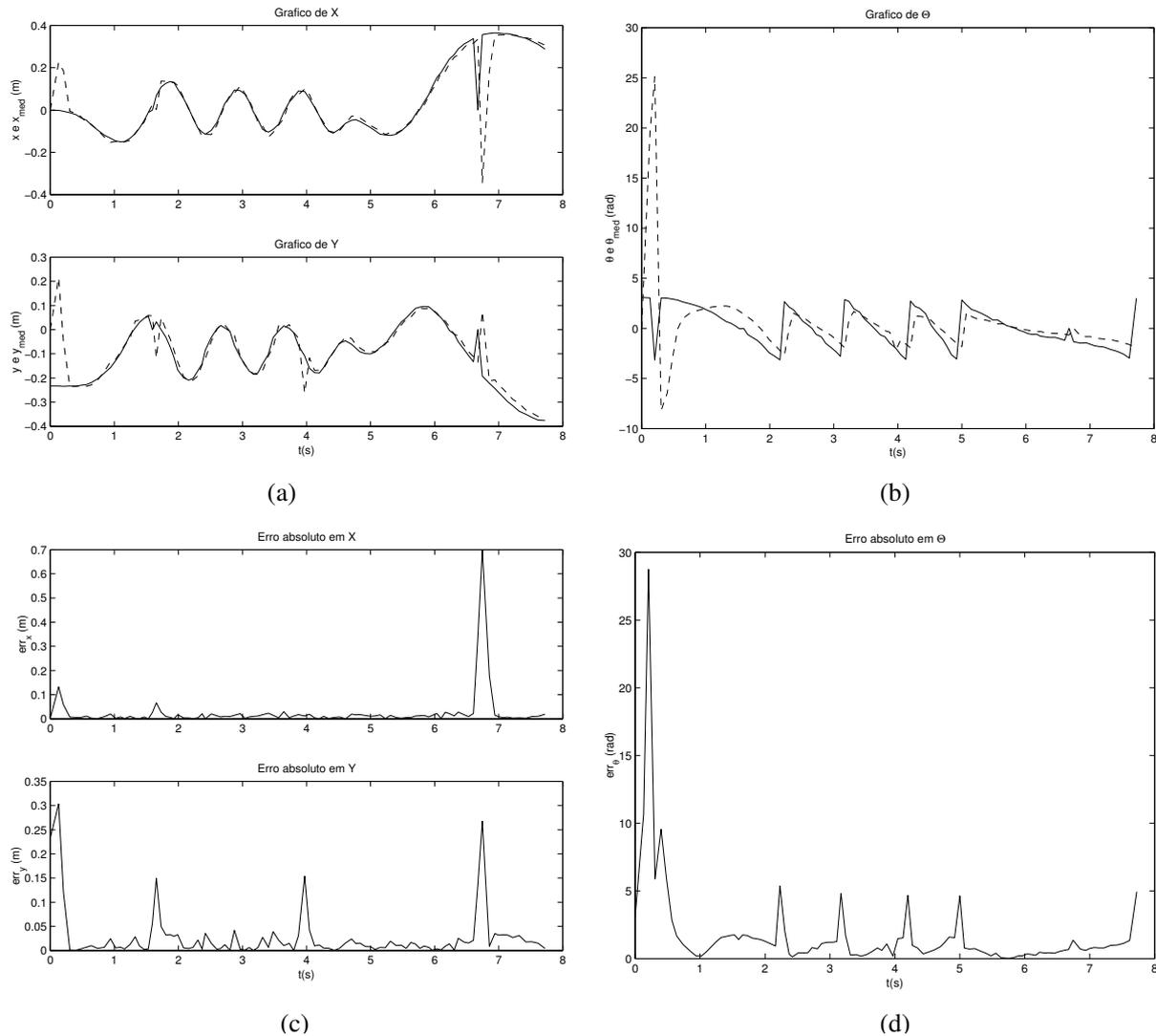


Figura 4.2: Valores medidos (—) e estimados (---) de x e y (a) e θ (b), com $n = 2$. Os respectivos erros absolutos são apresentados em (c) e (d)

de trajetória. Como exemplo, a figura 4.5 mostra o resultado da substituição dos valores estimados nos dados de configuração ausentes de x e y .

4.3.1 Normalização do Ângulo

Nos resultados apresentados anteriormente percebe-se a existência de erros de estimação da orientação θ do minirobô, o que compromete o desempenho dos módulos de estratégia e de controle.

Estes erros ocorrem porque o sistema de visão envia o valor do ângulo de orientação entre o intervalo fechado $[-\pi, \pi]$, fazendo com que a função que descreve θ seja não-linear. Para solucionar este problema optou-se por mudar a normalização do ângulo de orientação de $[-\pi, \pi]$ para $(-\infty, \infty)$. Esta mudança é feita iterativamente à medida que o ângulo é

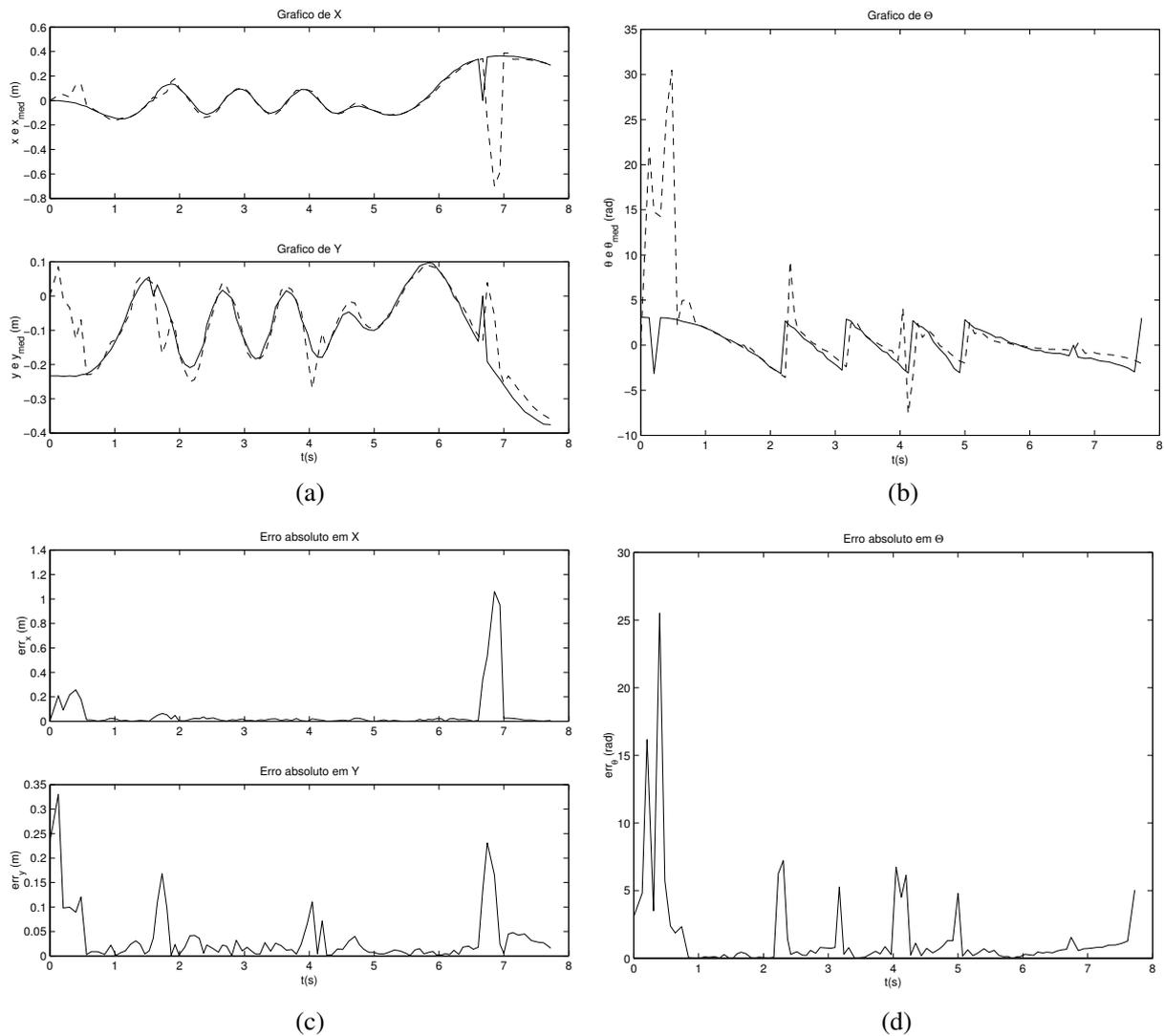


Figura 4.3: Valores medidos (—) e estimados (---) de x , y e θ mostrados em (a) e (b), com $n = 5$. Os respectivos erros absolutos são apresentados em (c) e (d)

fornecido pelo módulo de visão. A equação que rege esta normalização é:

$$\theta_k = \tan^{-1} \left(\frac{\sin \theta_k}{\cos \theta_k} \right)$$

Através desta normalização obtém-se resultados mais precisos para θ tornando a aplicação do algoritmo dos Mínimos Quadrados mais eficiente. A figura 4.6 mostra os valores estimados para θ no novo intervalo, com $n = 2$.

4.4 Conclusões

Este capítulo mostrou a etapa de estimação utilizada para os minirobôs do projeto do futebol do DCA-UFRN. Uma etapa de estimação tornou-se necessária por que, sob algumas condições, o sistema de visão não mede adequadamente os valores dos parâmetros de configuração fazendo com que outros módulos trabalhem com erros inviabilizando a

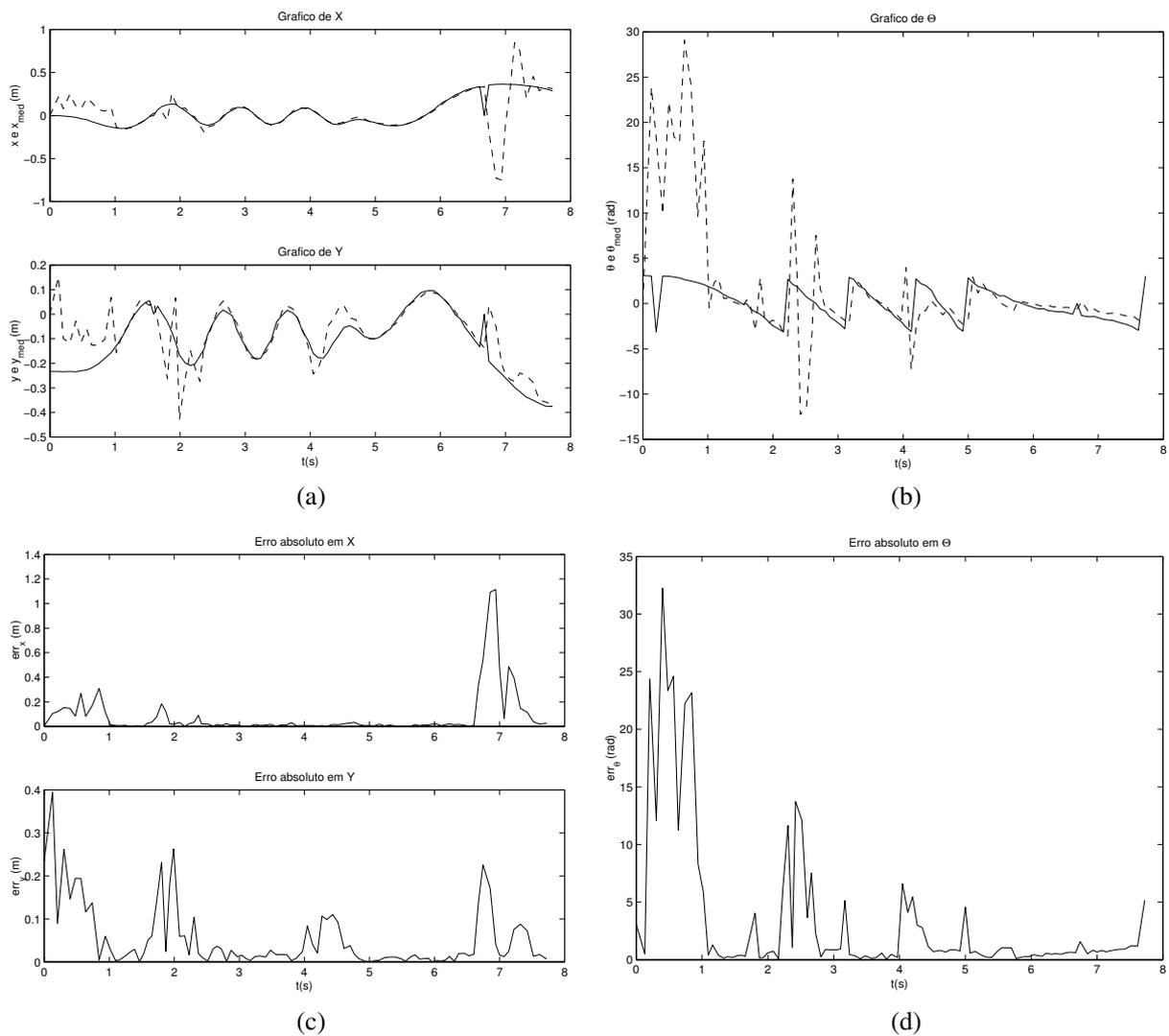


Figura 4.4: Valores medidos (—) e estimados (---) de x , y e θ , com $n = 10$, mostrados em (a) e (b), e os respectivos erros absolutos mostrados em (c) e (d)

eficácia do projeto como um todo. Optou-se por usar o algoritmo dos Mínimos Quadrados para a estimação da configuração dos minirobôs devido a sua comprovada eficiência em resolução de sistemas lineares. Pelos resultados apresentados percebeu-se que este método pode ser utilizado no problema encontrado, desde que sejam feitas as considerações sobre a linearização dos minirobôs. No próximo capítulo será apresentada a técnica adotada para encontrar caminhos admissíveis pelo minirobô.

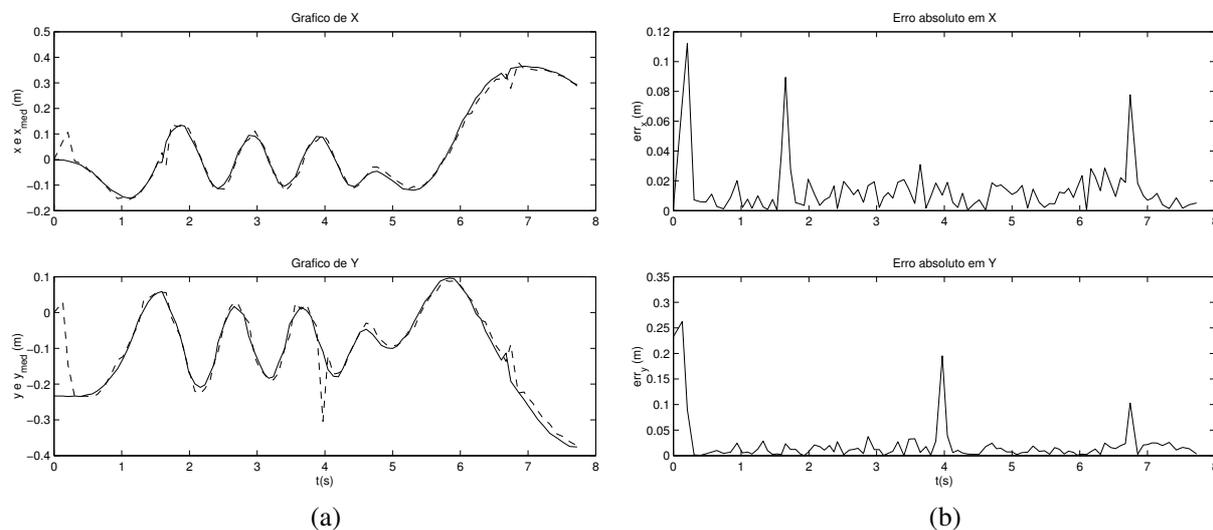


Figura 4.5: Utilização dos valores estimados (—) nos instantes de ausência dos valores medidos (—) de x e y (a) e os respectivos erros absolutos (b)

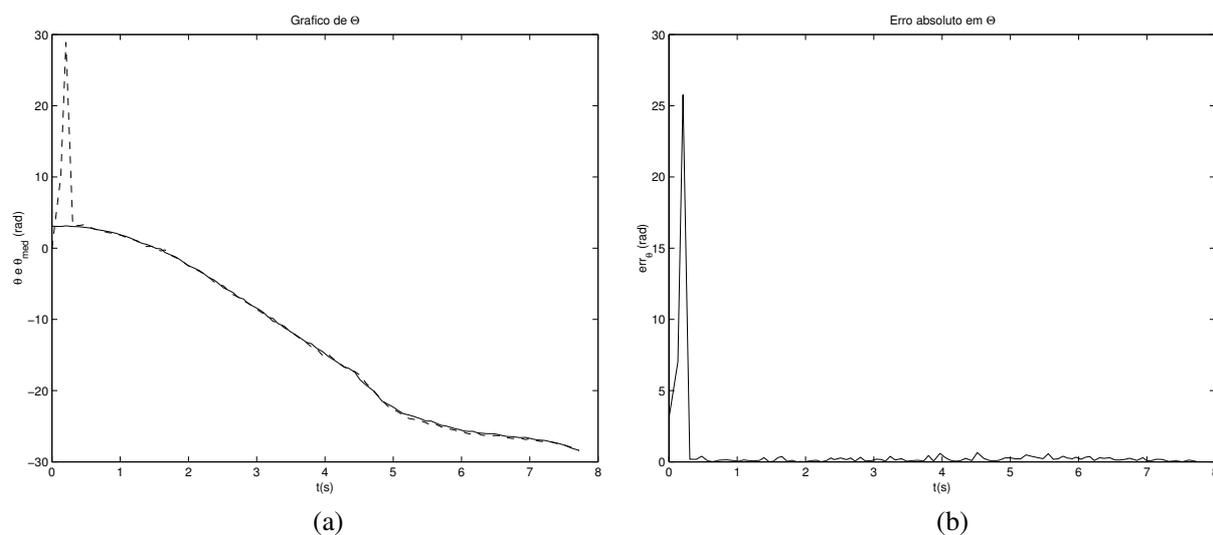


Figura 4.6: Valores medidos (—) e estimados (—) de θ após normalização, com $n = 2$, (a) e erro absoluto encontrado (b)

Capítulo 5

Planejamento de Caminhos para os Minirroboês

No capítulo anterior foi visto que para auxiliar o sistema de visão do projeto de futebol de robôs foi necessário implementar um módulo para estimar a configuração dos minirroboês de forma que, a cada passo de amostragem, as informações x , y e θ dos robôs fossem disponíveis para os outros módulos do projeto. Neste capítulo será apresentado o sistema elaborado para o planejamento de caminhos e método de geração de trajetória empregado, que resultam no gerador de referências para o controlador do minirroboê.

5.1 Motivação

Um módulo de planejamento de caminhos é necessário em qualquer robô que navegue de forma autônoma porque é responsável por encontrar um caminho que pode ser percorrido em um ambiente de trabalho. Como apresentado no capítulo 3, há várias técnicas para o planejamento de caminhos, entretanto nenhuma delas se aplica ao futebol de robôs devido o ambiente de trabalho ser dinâmico. Um outro problema é que o minirroboê é um sistema não-holonômico. Sistemas com este tipo de restrição percorrem apenas caminhos que possam ser diferenciáveis. Dessa forma, não é possível utilizar as técnicas de planejamento de caminho geométrico apresentadas por Latombe [Lat91], pois embora sejam métodos clássicos na literatura, elas não tratam de ambientes com obstáculos dinâmicos e robôs com restrições cinemáticas, como os de uma equipe de futebol, por exemplo.

Diversas propostas foram testadas para este fim. Com relação aos obstáculos, adotou-se que o desvio deverá ser realizado localmente e em tempo real. Assim, na presença de um obstáculo próximo ao caminho que o robô esteja percorrendo, uma nova configuração intermediária, afastada o suficiente do obstáculo, deve ser fornecida ao módulo de planejamento de caminhos e um novo caminho deve ser gerado passando por esta nova configuração e livrando, dessa forma, o robô da colisão. Com relação à geração do caminho propriamente dito, a idéia é utilizar diretamente algum método de adaptação de caminho geométrico, para dessa forma determinar um caminho realizável por um robô com restrição não-holonômica. As próximas seções apresentarão as técnicas estudadas, as dificuldades encontradas e o método adotado para a equipe do DCA-UFRN.

5.2 Abordagem Clássica

A abordagem clássica para criação de um caminho entre duas configurações distintas consiste em concatenar arcos de círculo e segmentos de reta. Esta foi a proposta de Reeds e Shepp [RS90] para um carro que se movimenta para frente e para trás e que possui limitação no raio de curvatura. Um exemplo de caminho é apresentado na figura 5.1.

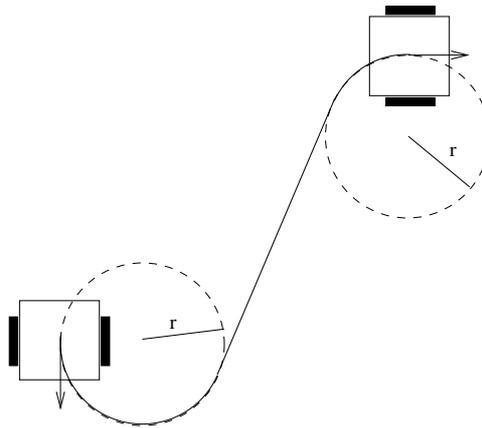


Figura 5.1: Caminho composto por dois arcos de círculo de raio r e um segmento de reta

Como todo sistema não-holonômico, os minirobôs do DCA-UFRN são capazes de percorrer tais caminhos, uma vez que eles se mostram diferenciáveis. Porém, para que o caminho seja o menor possível o raio do círculo considerado tem que ser mínimo, que no caso dos minirobôs é igual a zero. Quando aplicada para robôs com acionamento diferencial, como os minirobôs, a abordagem clássica divide o planejamento de caminho em três fases distintas, que são:

1. uma rotação em torno do próprio eixo até apontar para a posição final desejada;
2. um movimento em linha reta da posição inicial até alcançar a posição final; e
3. outro movimento de rotação para alcançar a orientação final desejada.

Este método de planejamento permite que os caminhos encontrados respeitem a restrição não-holonômica do minirobô, embora ele possua algumas desvantagens. Primeiramente, eles não são realizáveis com um único movimento contínuo porque isso implicaria em acelerações infinitas nas transições de fases. Outra desvantagem é que como a abordagem clássica leva em consideração a configuração final do robô, que na maioria das vezes é um alvo móvel, e devido a dinâmica do futebol, pode ocorrer que o robô jamais saia da primeira fase e permaneça girando indefinidamente tentando apontar para o alvo. Além disso, somente é possível alterar o movimento do robô quando as três fases forem cumpridas. Devido a estes problemas, buscou-se uma técnica que não dividisse o movimento do robô em fases, implicando assim em tornar suas velocidades linear e angular iguais a zero, e que continuasse a respeitar a restrição cinemática do minirobô.

5.3 Aproximação por Elipse

A segunda proposta adotada foi a aproximação do caminho do minirobô entre duas configurações distintas por um arco de elipse, como apresentado na figura 5.2.

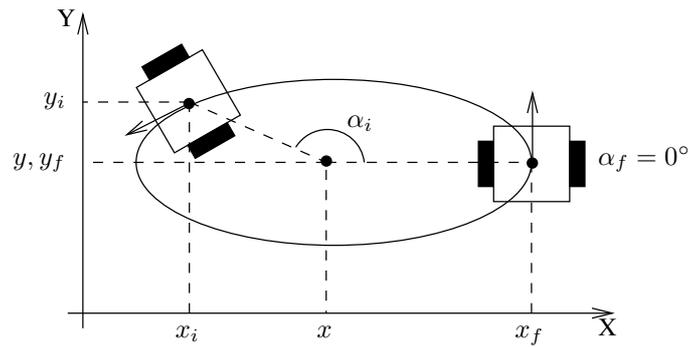


Figura 5.2: Caminho do minirobô aproximado por um arco de elipse

A idéia é apropriada pois os caminhos encontrados são diferenciáveis e, portanto, realizáveis pelo minirobô. A definição do problema é: dadas as configurações inicial e final, respectivamente (x_i, y_i, θ_i) e (x_f, y_f, θ_f) , encontrar uma elipse que passe pelos pontos (x_i, y_i) e (x_f, y_f) com $\tan \theta_i$ e $\tan \theta_f$ como as respectivas tangentes.

Sendo a equação de uma elipse dada por:

$$\left(\frac{x - x_c}{a}\right)^2 + \left(\frac{y - y_c}{b}\right)^2 = 1$$

para modelar o problema é preciso escrever a equação paramétrica de x e y . Assim, pela figura 5.3 tem-se que este modelo é dado pela equação 5.1, onde α é o ângulo formado entre o centro do robô e o centro da elipse em relação ao eixo x desta elipse.

$$\begin{cases} x = x_c + a \cos \alpha \\ y = y_c + b \sin \alpha \\ \theta = \tan^{-1} \left(\frac{dx/d\alpha}{dy/d\alpha} \right) = \tan^{-1} \left(\frac{a \sin \alpha}{-b \cos \alpha} \right) \end{cases} \quad (5.1)$$

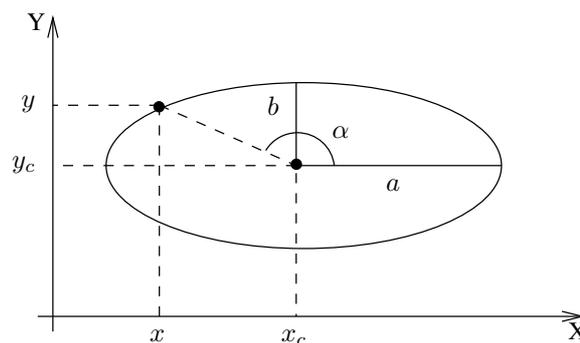


Figura 5.3: Parâmetros característicos de uma elipse

Substituindo os valores das configurações inicial e final na equação 5.1, tem-se o sistema dado pela equação 5.2. Percebe-se que há solução matemática para este problema pois têm-se seis incógnitas $(x_c, y_c, a, b, \alpha_i, \alpha_f)$ e seis equações. Entretanto, não é um sistema de trivial solução pois as equações envolvidas são não-lineares. Diante disto, buscou-se outra abordagem para o planejamento do caminho, optando-se por ligar as configurações inicial e final através de polinômios paramétricos.

$$\begin{cases} x_i = x_c + a \cos \alpha_i \\ y_i = y_c + b \sin \alpha_i \\ \theta_i = \tan^{-1} \left(\frac{a \sin \alpha_i}{-b \cos \alpha_i} \right) \\ x_f = x_c + a \cos \alpha_f \\ y_f = y_c + b \sin \alpha_f \\ \theta_f = \tan^{-1} \left(\frac{a \sin \alpha_f}{-b \cos \alpha_f} \right) \end{cases} \quad (5.2)$$

5.4 Polinômios Paramétricos

O problema básico do planejamento de caminhos é a união de uma configuração inicial com uma final através de um conjunto de configurações, ou caminho, de forma a atender a restrição não-holonômica do minirobô. Diante deste problema e das desvantagens das abordagens apresentadas anteriormente, optou-se gerar um caminho onde as variáveis de configuração x e y são definidas por um polinômio de parâmetro λ , adimensional e não relacionado com o tempo. O ângulo de orientação θ , devido à restrição não-holonômica, não é calculado mas sim imposto através da relação entre as derivadas em relação a λ de x e y . A faixa de valores deste parâmetro varia entre $(0, 1)$ de tal forma que quando $\lambda = 0$, o minirobô está em sua configuração inicial e quando $\lambda = 1$, o minirobô encontra-se na sua configuração final.

5.4.1 Polinômios de Segundo Grau

Definindo as variáveis x e y como polinômios de segundo grau, têm-se:

$$\begin{aligned} x(\lambda) &= a_0 + a_1\lambda + a_2\lambda^2 \\ y(\lambda) &= b_0 + b_1\lambda + b_2\lambda^2 \end{aligned} \quad (5.3)$$

Como é imposto que o ângulo de orientação θ do minirobô seja sempre tangente à trajetória que está sendo executada, a orientação é dada por:

$$\theta(\lambda) = \tan^{-1} \left(\frac{dy/d\lambda}{dx/d\lambda} \right) = \tan^{-1} \left(\frac{b_1 + 2b_2\lambda}{a_1 + 2a_2\lambda} \right) \quad (5.4)$$

Como forma de simplificação, trabalha-se com a função $d(\lambda)$, definida como:

$$d(\lambda) = \frac{b_1 + 2b_2\lambda}{a_1 + 2a_2\lambda}$$

Assim, o ângulo de orientação pode ser escrito como:

$$\theta(\lambda) = \tan^{-1}(d(\lambda))$$

Para $\lambda = 0$ e $\lambda = 1$ têm-se respectivamente:

$$\begin{aligned} (x(0), y(0)) &= (x_i, y_i) & (x(1), y(1)) &= (x_f, y_f) \\ d(0) &= d_i = \tan \theta_i & d(1) &= d_f = \tan \theta_f \end{aligned}$$

Aplicando estas condições de contorno nas equações 5.3 e 5.4, ter-se-á um sistema linear de fácil solução com seis incógnitas e seis equações, dada por:

$$\begin{aligned} b_1 + 2b_2 - d_f a_1 - 2d_f a_2 &= 0 \\ a_0 + a_1 + a_2 &= x_f \\ b_0 + b_1 + b_2 &= y_f \\ b_1 &= d_i a_1 \\ a_0 &= x_i \\ b_0 &= y_i \end{aligned}$$

Resolvendo este sistema tem-se:

$$\begin{cases} a_0 = x_i \\ a_1 = \frac{2(d_f \Delta x - \Delta y)}{d_f - d_i} \\ a_2 = \Delta x - a_1 \\ b_0 = y_i \\ b_1 = d_i a_1 \\ b_2 = \Delta y - b_1 \end{cases}$$

onde $\Delta x = x_f - x_i$ e $\Delta y = y_f - y_i$.

Portanto, com estes valores para os coeficientes das equações 5.3 garante-se que entre quaisquer duas configurações inicial e final existirá um caminho que as una. Isto será exceção nas singularidades matemáticas que ocorrem quando $\tan \theta_i = \tan \theta_f$ ou quando θ_i e/ou θ_f forem iguais a $\pm\pi/2$, o que representa uma desvantagem do método.

As figuras 5.4 mostram alguns caminhos obtidos com esta abordagem. Pode-se perceber que para algumas configurações inicial e final o caminho encontrado mostra-se adequado para um sistema com acionamento diferencial. Entretanto, para outras configurações o caminho força a realização de um movimento desnecessário em relação à configuração final. Esta desvantagem pode ser significativa no ambiente de futebol de robôs, onde se deseja que o robô alcance rapidamente a bola.

Diante destas desvantagens encontradas optou-se por utilizar polinômios de terceiro grau, o que não retira a originalidade do método proposto. Ao adicionar mais um grau aos polinômios torna-se possível adequar os caminhos segundo algum critério, devido a presença de coeficientes que podem ter seus valores arbitrados.

5.4.2 Polinômios de Terceiro Grau

As variáveis de configuração $x(\lambda)$, $y(\lambda)$ e o ângulo $\theta(\lambda)$ são dados por:

$$\begin{aligned} x(\lambda) &= a_0 + a_1 \lambda + a_2 \lambda^2 + a_3 \lambda^3 \\ y(\lambda) &= b_0 + b_1 \lambda + b_2 \lambda^2 + b_3 \lambda^3 \\ \theta(\lambda) &= \tan^{-1} \left(\frac{b_1 + 2b_2 \lambda + 3b_3 \lambda^2}{a_1 + 2a_2 \lambda + 3a_3 \lambda^2} \right) \end{aligned}$$

O parâmetro λ , sem relação com o tempo, varia de 0 a 1, sendo (x_i, y_i, θ_i) para $\lambda = 0$ e (x_f, y_f, θ_f) para $\lambda = 1$. Usando estas condições de contorno e efetuando algumas

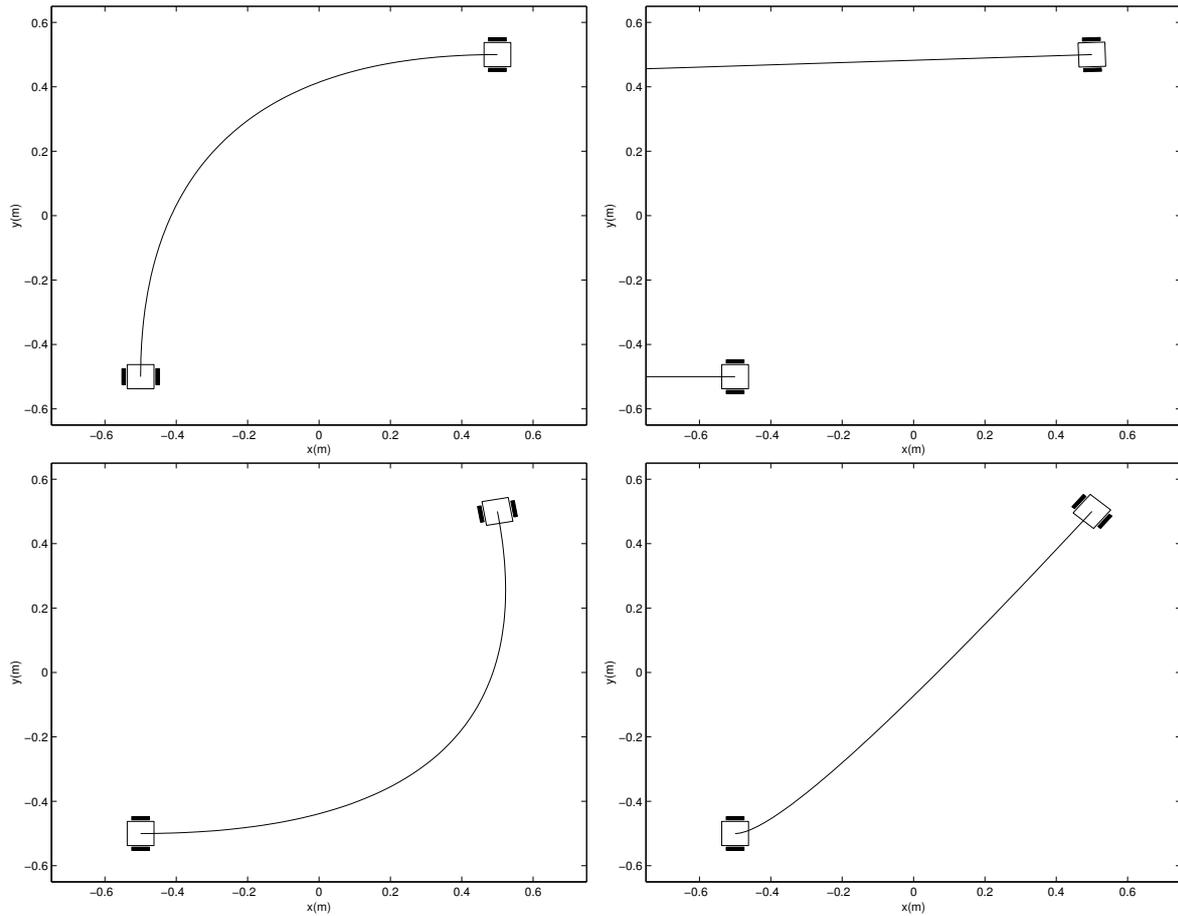


Figura 5.4: Conjunto de caminhos encontrados através da utilização de polinômios de segundo grau nas variáveis x e y de configuração

operações algébricas, chega-se a um sistema dado pela equação matricial 5.5. Todo o desenvolvimento matemático desta abordagem encontra-se no apêndice B.

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3d_f & 2d_f & d_f \\ 0 & 0 & 1 & 0 & 0 & -d_i \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ a_3 \\ a_2 \\ a_1 \end{bmatrix} = \begin{bmatrix} \Delta y \\ 3\Delta y \\ 0 \\ \Delta x \end{bmatrix} \quad (5.5)$$

com $\Delta x = x_f - x_i$, $\Delta y = y_f - y_i$, $d_i = \tan \theta_i$ e $d_f = \tan \theta_f$.

O sistema 5.5 é definido para as situações em que θ_i e θ_f são diferentes de $\pm\pi/2$. Vê-se que neste sistema têm-se quatro equações e seis parâmetros. Assim, arbitra-se os valores de dois dos seis parâmetros. Tomando-se a_1 e a_2 como tais parâmetros livres e efetuando

as operações necessárias, chega-se a:

$$\begin{cases} a_0 = x_i \\ a_1 \rightarrow \text{valor arbitrado} \\ a_2 \rightarrow \text{valor arbitrado} \\ a_3 = \Delta x - a_2 - a_1 \\ b_1 = d_i a_1 \\ b_2 = 3\Delta y - 3d_f \Delta x + d_f a_2 - 2(d_i - d_f) a_1 \\ b_3 = 3d_f \Delta x - 2\Delta y - d_f a_2 - (2d_f - d_i) a_1 \end{cases}$$

Os parâmetros a_1 e a_2 podem assumir quaisquer valores reais. As figuras 5.5 mostram os caminhos encontrados através deste método arbitrando os valores $a_1 = 1$ e $a_2 = 2$ para os coeficientes livres.

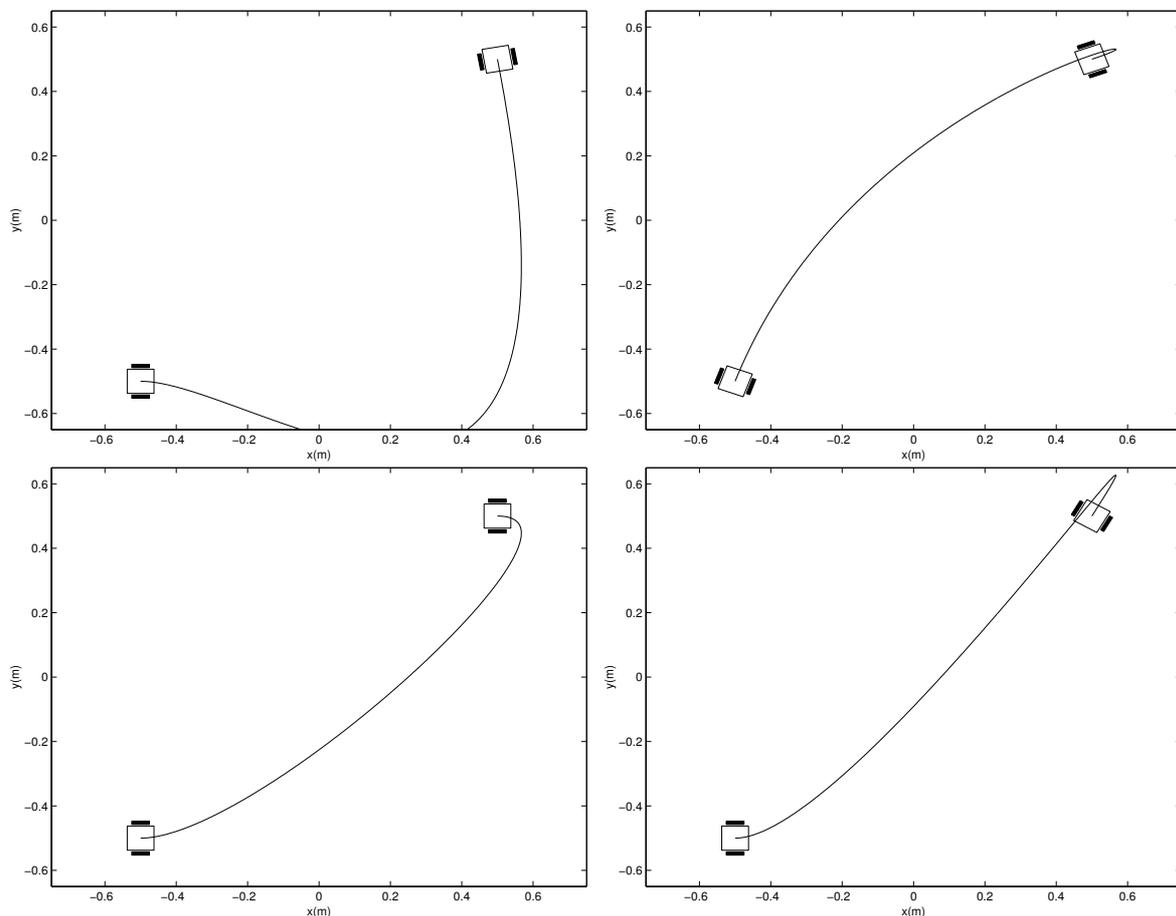


Figura 5.5: Caminhos encontrados através de polinômios de terceiro grau nas variáveis de configuração x e y

Comparando-se estes gráficos apresentados com os obtidos da função de segundo grau percebe-se que os resultados não são satisfatórios. Isto pode ser explicado pelos valores atribuídos aos coeficientes livres que não são apropriados para produzirem um caminho adequado. Isto implica que eles devem ser escolhidos de acordo com algum critério, de forma que os caminhos resultantes possam ser utilizados no ambiente de futebol de robôs.

5.5 Adequação dos Coeficientes Livres

A proposta do método dos polinômios de terceiro grau é gerar um caminho adequado para o minirobô, entre duas configurações distintas. Pelos exemplos apresentados na figura 5.5, percebeu-se que, embora respeitem a restrição não-holonômica, nem sempre é possível obter um bom caminho, pela ótica do bom senso. Assim, para obter caminhos mais “inteligentes” é necessário escolher valores apropriados para os coeficientes livres. Dessa forma, torna-se necessário estabelecer um critério para a determinação do caminho. Isto implicará que os coeficientes livres sejam escolhidos a partir de uma faixa de valores a ser determinada.

Há vários critérios que poderiam ser adotados tais como o menor caminho, ou a trajetória que fosse realizada em menor tempo. Entretanto, estes critérios normalmente recaem em movimentos de rotação seguidos de movimentos em linha reta, uma vez que o menor caminho entre dois pontos é uma reta. O critério adotado foi considerado o mais adequado para robôs com acionamento diferencial. Ele consiste em impedir que os polinômios $x(\lambda)$ e $y(\lambda)$ tenham valores máximo ou mínimo entre $0 < \lambda < 1$. Como consequência o caminho fica confinado, sempre que possível, em um quadrilátero delimitado pelas posições inicial e final do minirobô no ambiente de trabalho. Se isto ocorrer, é garantido que o caminho fica limitado ao ambiente de trabalho, o que pode ser utilizado para verificar prováveis colisões.

Com a utilização deste critério, tornou-se possível calcular os valores mais adequados dos coeficientes livres para obter o caminho desejado. Assim, para que o caminho não apresente valores de máximo ou mínimo entre o intervalo $[0, 1]$, as raízes das derivadas em relação à λ de $x(\lambda)$ e $y(\lambda)$ têm que ser:

1. reais e maiores ou iguais a 1; ou
2. reais e menores ou iguais a 0; ou
3. reais com uma raiz maior ou igual a 1, e outra menor ou igual a 0; ou
4. complexas.

Estas condições fazem com que sejam encontrados sistemas de inequações de primeiro grau que possibilitam achar os valores desejados para os coeficientes livres. Os valores destes coeficientes são determinados através de regiões no espaço dos coeficientes livres, tanto para x quanto para y . O desenvolvimento matemático descrito é melhor detalhado no apêndice B.

Como resultado final, para que não haja valores de máximos ou mínimos em x , no intervalo $(0, 1)$ de λ , os coeficientes livres a_1 e a_2 precisam estar contidos em uma região delimitada pelas seguintes inequações:

$$\begin{cases} a_1 \geq 0 \\ a_2 \geq -a_1 \\ a_2 \leq 3\Delta x - 2a_1 \end{cases}$$

que pode ser visualizada na figura 5.6, quando Δx for considerado positivo.

A região final para $y(\lambda)$ pode ser expressada pelo sistema de inequações 5.6. Entretanto, como a região para atender aos critérios na direção y depende de Δx , Δy , d_i e d_f , ela pode apresentar-se de forma não coincidente com a região para atender os critérios na direção x .

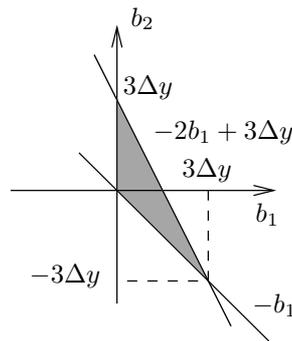


Figura 5.6: Região final admissível para $x(\lambda)$

$$\begin{cases} d_i a_1 \geq 0 \\ d_f a_2 \geq (d_i - 2d_f)a_1 + 3(df\Delta x - \Delta y) \\ d_f a_2 \leq 3d_f\Delta x - 2d_f a_1 \end{cases} \quad (5.6)$$

Considerando Δx , Δy , d_i e d_f positivos, chega-se à região mostrada na figura 5.7.

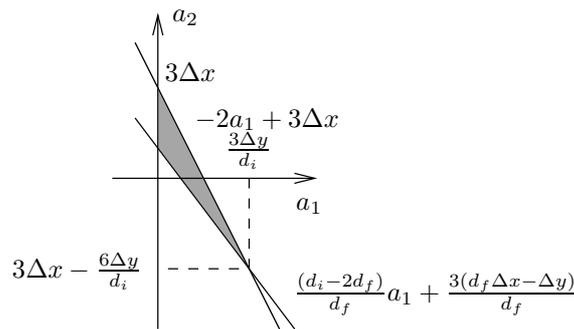


Figura 5.7: Região admissível para $y(\lambda)$

Para a realização dos testes, foram escolhidos os valores $a_1 = 0$ e $a_2 = \Delta x$, que corresponde ao centróide da região admissível encontrada para x (figura 5.6). A figura 5.8 mostra os caminhos obtidos com este método.

Pelos resultados apresentados vê-se que a abordagem de polinômios paramétricos de terceiro grau para determinar as variáveis de configuração do robô são adequados para uma vasta gama de configurações. Os caminhos gerados, em sua maioria, mostraram-se diferenciáveis o que atende ao requisito estabelecido, ou seja, resulta em caminhos que podem ser percorridos por um robô com duas rodas sujeito à restrição não-holonômica. As desvantagens obtidas foram a perda do sentido de chegada, ou seja, do ângulo final, uma vez que as equações paramétricas trabalham com a tangente dos ângulos. Outra desvantagem é que, para algumas configurações, os caminhos não são determinados, como por exemplo caminhos onde não há deslocamento no eixo x .

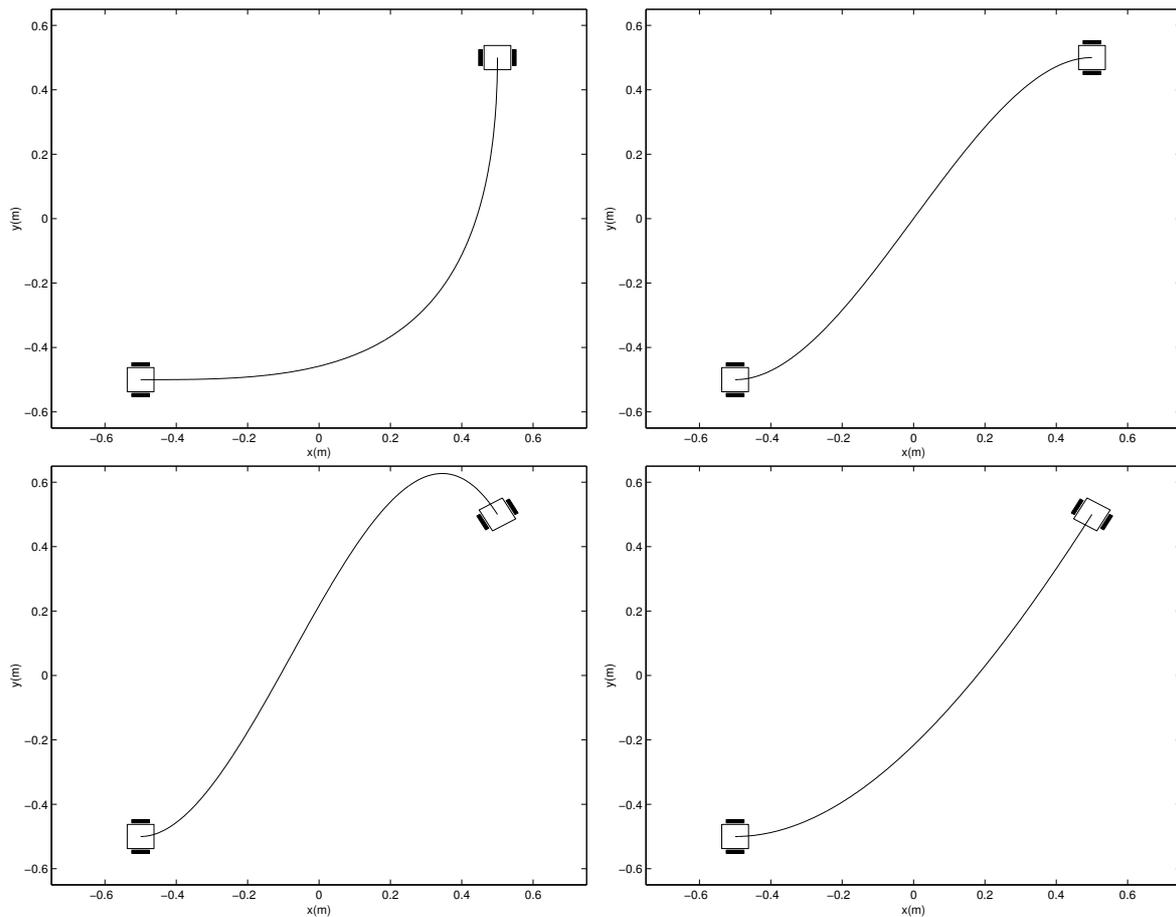


Figura 5.8: Caminhos encontrados através da utilização de polinômios de terceiro grau nas variáveis x e y , com os valores apropriados para os coeficientes livres

5.6 Geração da Trajetória

Os resultados apresentados nas seções anteriores tratavam a variável λ como independente do tempo, ou seja, os resultados apresentados não se configuravam como uma trajetória, e sim como um caminho. O planejamento de trajetória, assim como qualquer etapa realizada no sistema de futebol dos minirobôs do DCA-UFRN têm que ser realizado a cada período de amostragem do sistema de visão, que é de $30ms$. Para criar esta trajetória tem-se que levar em consideração este tempo de amostragem e as possíveis velocidades desenvolvidas pelo robô.

Para gerar a trajetória é preciso calcular a distância entre a posição atual e final do robô. Duas propostas foram investigadas. A primeira era tomar a distância euclidiana entre as duas posições, o que foi descartado pois incluiria erros grosseiros de aproximação do comprimento do caminho encontrado. A segunda proposta é calcular o comprimento s do caminho real através de:

$$s(\lambda) = \int_0^1 \left[\sqrt{\left(\frac{dx}{d\delta}\right)^2 + \left(\frac{dy}{d\delta}\right)^2} \right] d\delta$$

Como o cálculo desta integral não é trivial utilizou-se uma aproximação do compri-

mento do caminho $s(\lambda)$ através da expansão pela série de Taylor. Assim, tem-se:

$$s(\lambda) = s(0) + \frac{\dot{s}(0)}{1!}\lambda + \frac{\ddot{s}(0)}{2!}\lambda^2 + \frac{\overset{\cdot\cdot}{s}(0)}{3!}\lambda^3 + \dots$$

Sendo $0 \leq \lambda \leq 1$, pode-se truncar a equação acima no quarto termo, uma vez que λ^4 é aproximadamente 0. Pode-se considerar também que no início do movimento o comprimento do caminho é nulo, ou seja, $s(0) = 0$. Dessa forma, tem-se:

$$s(\lambda) = \frac{\beta_1}{1!}\lambda + \frac{\beta_2}{2!}\lambda^2 + \frac{\beta_3}{3!}\lambda^3$$

onde:

$$\begin{cases} \beta_1 = \sqrt{a_1^2 + b_1^2} \\ \beta_2 = 2 \frac{a_1 a_2 + b_1 b_2}{\sqrt{a_1^2 + b_1^2}} \\ \beta_3 = 2 \frac{[2(a_2^2 + b_2^2) + 3(a_1 a_3 + b_1 b_3)]}{\sqrt{a_1^2 + b_1^2}} - 4 \frac{(a_1 a_2 + b_1 b_2)^2}{(a_1^2 + b_1^2) \sqrt{a_1^2 + b_1^2}} \end{cases}$$

Definindo $\alpha = v_r \cdot \Delta t$ como a distância desejada que o minirobô deve percorrer durante o instante de amostragem seguinte, sendo v_r a velocidade média do robô e Δt o período de amostragem do sistema de visão, o problema da geração de trajetória é encontrar o valor de λ para que o minirobô esteja a uma distância α de sua posição atual, no próximo intervalo de amostragem. Em outras palavras, deve-se encontrar λ para satisfazer a seguinte equação:

$$s(\lambda) - \alpha = 0$$

Esta equação pode ser solucionada através de qualquer método numérico para resolução de equações não-lineares, como por exemplo o método da bisseção. Como o comprimento $s(\lambda)$ é monotonicamente crescente, a solução pode ser obtida em poucos laços o que torna esta técnica de geração de trajetórias adequada para ser usada em tempo real. O valor λ encontrado é utilizado para gerar a configuração de referência através das variáveis de configuração $x(\lambda)$, $y(\lambda)$ e $\theta(\lambda)$ do minirobô.

Um simples algoritmo para a geração das trajetórias pode ser elaborado para a melhor compreensão desta técnica:

1. enquanto o minirobô não estiver na sua posição final, ou a uma mínima distância satisfatória dela, deve-se:
 - (a) encontrar os coeficientes dos polinômios de terceiro grau $x(\lambda)$ e $y(\lambda)$ através do método apresentado na seção 5.4.2;
 - (b) encontrar o valor de λ , através de algum método de resolução de equações não-lineares (Bisseção, Newton-Raphson, Método da Secante, etc.) de forma que o minirobô possa estar a uma distância α de sua posição atual no próximo período de amostragem;
 - (c) encontrar, por fim, o valor de x , y , θ para o λ calculado. Estes valores serão a referência para o controlador do minirobô.

Como exemplo do funcionamento deste método, a figura 5.9 mostra três trajetórias executadas pelo robô no espaço de trabalho e o comportamento das variáveis de configuração x , y e θ ao longo do tempo. A configuração inicial é igual a $(0, 0, 0, 0, 0, 0)$ e a configuração final consistiu da posição $(0, 5, 0, 5)$ e das três orientações $5\pi/18$, 0 e $-5\pi/18$. Para este exemplo adotou-se um período de amostragem igual a $0,033s$ e uma distância α igual a $0,0165m$, que corresponde a uma velocidade média igual a $0,5m/s$. As referências para o controlador do minirobô são marcadas com o símbolo \circ e sobrepõem exatamente o caminho original entre as configurações inicial e final. O espaçamento parcialmente regular entre estas referências permitem que o robô possa desenvolver uma velocidade média constante durante o seu movimento.

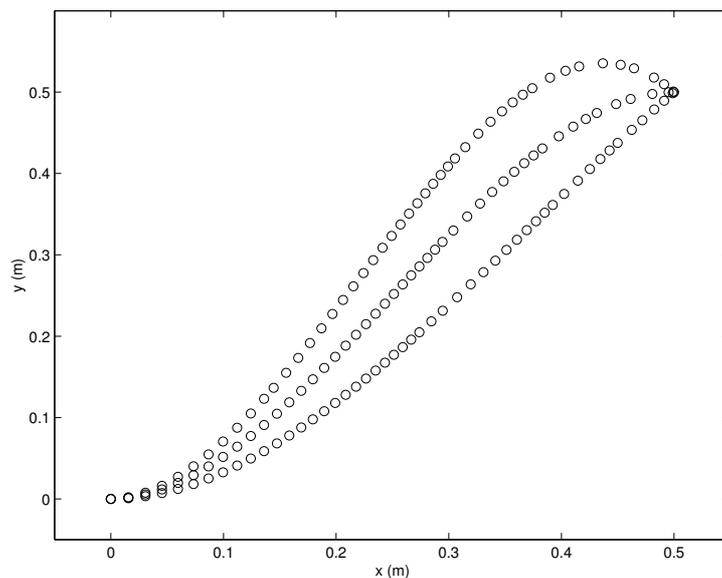


Figura 5.9: Trajetórias calculadas para serem desenvolvidas pelo minirobô

5.7 Conclusões

Este capítulo apresentou a técnica de planejamento de caminhos a ser utilizada na frota de minirobôs móveis do DCA-UFRN. O método de planejamento proposto consiste em unir as configurações inicial e final por um caminho determinado através de polinômios paramétricos de terceiro grau. O principal objetivo do método é obter caminhos diferenciáveis e, conseqüentemente, contínuos. Isto é um requisito para que um robô móvel com duas rodas e sujeito a uma restrição não-holonômica possa percorrê-lo. As variáveis de configuração x e y são determinadas por polinômios paramétricos de terceiro grau. A orientação θ , devido à restrição cinemática, é imposta ser igual ao arco-tangente da relação entre as derivadas em relação à λ de x e y .

Para que este método fosse utilizado, outras abordagens foram analisadas. A primeira técnica foi a abordagem clássica de Reeds e Shepp [RS90] que gera caminhos formados por arcos de círculo de raio mínimo e segmentos de reta. Como o menor raio de rotação para o minirobô é zero, o movimento de rotação do robô seria sobre o próprio eixo. Este método gera movimentos que são realizáveis pelo minirobô abordado mas tem a desvantagem de separar a trajetória em fases pois resume o movimento em um giro, um deslocamento em

linha reta e outro giro final. O segundo método foi aproximar o caminho por um segmento de uma elipse. Esta abordagem também gera caminhos que podem ser percorridos pelos robôs, mas utiliza equações não-lineares, cuja determinação analítica de suas soluções é não-trivial. A terceira abordagem foi utilizar polinômios paramétricos de segundo grau para determinar x , y e θ . Alguns resultados foram apresentados. A maioria mostrou-se satisfatória mas houve situações em que não se alcançou um caminho desejado, o que foi resolvido ao utilizar polinômios de terceiro grau.

Este último método apresentou vantagens sobre os demais. Além disso, conseguiu-se graus de liberdade suficiente para aprimorar o caminho nas direções x e y , de modo a torná-lo apropriado para o minirobô e a sua aplicação. Os resultados apresentados mostraram a eficácia da abordagem, o que garante que este método pode ser utilizado no futebol de robôs do DCA-UFRN. A desvantagem encontrada foi perder o significado do sentido de chegada, uma vez que trabalha-se com as tangentes dos ângulos além de não encontrar caminhos que impliquem somente no deslocamento de um eixo.

Foi apresentado também um método de gerar a trajetória para o minirobô. Ele consiste em determinar o comprimento do caminho através da expansão por série de Taylor e encontrar o valor de λ de forma que o robô possa percorrer uma distância pré-determinada α , levando-se em consideração o período de amostragem do sistema de visão e a velocidade média executada pelos minirobôs.

Ressalta-se novamente que a vantagem dos métodos aqui apresentados são a simplicidade e a sua adequabilidade para ser usado em sistemas robóticos que trabalham em ambientes dinâmicos e que necessitam recalcular, em todo os instantes de amostragem, o seu caminho e trajetória em direção ao alvo. O próximo capítulo mostrará os resultados obtidos em um sistema simulado de movimento para os robôs em um ambiente de futebol. Sendo este simulador uma representação do sistema real, uma técnica de controle simples que foi empregada para garantir os resultados e a validação dos métodos de planejamento de caminho e de trajetória adotados.

Capítulo 6

Resultados Experimentais

No capítulo 3 mostrou-se que o conjunto formado pelo planejamento de caminhos e geração de trajetória juntamente com o controle em baixo nível pode ser encarado como o sistema de controle do minirobô. Este capítulo apresentará os resultados obtidos da utilização da estratégia de planejamento de caminhos e geração de trajetória, apresentados no capítulo 5, em um minirobô simulado. Para testar a aplicabilidade das técnicas propostas, utilizou-se um sistema de controle simples capaz de realizar o rastreo da trajetória do robô. Os resultados foram obtidos em um simulador realístico cuja descrição também é apresentada neste capítulo, assim como a modelagem matemática do minirobô, onde da qual foi desenvolvido o simulador em questão.

6.1 Modelagem Matemática do Minirobô

A modelagem matemática do minirobô é importante pois fornece um conjunto de equações capaz de descrever o seu comportamento físico. O minirobô, que foi descrito de maneira introdutória no capítulo 3, consiste de um sistema móvel de duas rodas acionado por dois motores de corrente contínua, sendo um motor para cada roda. No minirobô não há sistema inteligente embarcado, ou seja, todo o processamento é realizado remotamente pelo computador que controla a equipe. A figura 6.1 mostra o diagrama de forças e torques que agem sobre o minirobô, cuja 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;
- $v_{d,e}$ são as velocidades lineares de um ponto nas bordas externas das rodas direita e esquerda, respectivamente;
- v é a velocidade linear do robô; e
- ω é a velocidade angular do robô.

A velocidade linear v e angular ω do minirobô são dadas por:

$$v = \frac{v_d + v_e}{2} \qquad \omega = \frac{v_d - v_e}{d}$$

Sendo as velocidades lineares das rodas dadas por $v_d = \omega_d r_d$ e $v_e = \omega_e r_e$, é possível encontrar as velocidades angulares das rodas do minirobô em função das velocidades

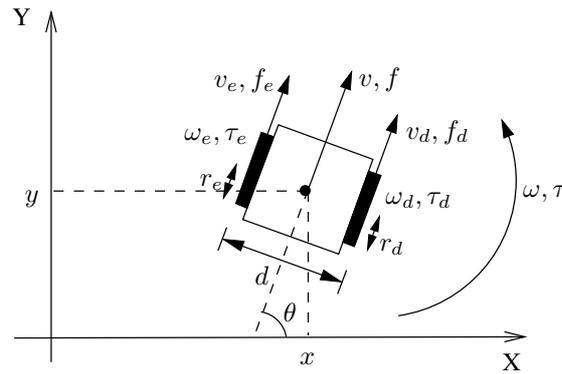


Figura 6.1: Diagrama das forças e torques que atuam sobre o minirobô

linear e angular. Assim, têm-se:

$$\omega_d = \frac{1}{r_d}v + \frac{d}{2r_d}\omega \quad \omega_e = \frac{1}{r_e}v - \frac{d}{2r_e}\omega$$

que na forma matricial pode ser escrita como:

$$\boldsymbol{\omega} = {}^\omega T_v \mathbf{v} \quad (6.1)$$

ou na forma:

$$\mathbf{v} = {}^\omega T_v^{-1} \boldsymbol{\omega} = {}^v T_\omega \boldsymbol{\omega} \quad (6.2)$$

onde:

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_d \\ \omega_e \end{bmatrix} \quad \mathbf{v} = \begin{bmatrix} v \\ \omega \end{bmatrix} \quad {}^\omega T_v = \begin{bmatrix} 1/r_d & d/2r_d \\ 1/r_e & -d/2r_e \end{bmatrix}$$

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 pode ser expresso por restrições matemáticas que se enquadram na categoria das restrições não-holonômicas. Considerando as componentes da velocidade linear nos eixos x e y , e a variação no tempo da orientação do robô, têm-se:

$$\begin{aligned} \dot{x} &= v \cos \theta \\ \dot{y} &= v \sin \theta \\ \dot{\theta} &= \omega \end{aligned} \quad (6.3)$$

Igualando-se o valor de v nas equações 6.3, chega-se à restrição não-holonômica:

$$\dot{x} \sin \theta = \dot{y} \cos \theta$$

As equações 6.3 podem ser escritas na forma matricial:

$$\dot{\mathbf{q}} = {}^q T_v \mathbf{v} \quad (6.4)$$

onde:

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

O vetor $\dot{\mathbf{q}}$ representa as velocidades no espaço de configurações. Substituindo-se a equação 6.2 em 6.4, pode-se obter a relação entre as velocidades no referencial dos atuadores e no referencial das variáveis de configuração:

$$\dot{\mathbf{q}} = {}^qT_v {}^vT_\omega \boldsymbol{\omega}$$

Pela definição da matriz qT_v , constata-se que:

$${}^qT_v {}^qT_v = I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Considerando que I é uma matriz identidade, pode-se obter o vetor \mathbf{v} em função das derivadas das variáveis de configuração. Assim, multiplicando a equação 6.4 por ${}^qT_v^T$, obtém-se:

$$\mathbf{v} = {}^qT_v^T \dot{\mathbf{q}}$$

6.1.1 Esforços sobre o Robô

Com as relações no referencial de atuadores, é possível encontrar relações entre os análogos elétricos da parte linear e da parte angular do minirobô com os análogos elétricos dos motores direito e esquerdo. A figura 6.2 mostra os análogos elétricos dos comportamentos linear e angular do minirobô. As equações derivadas destes circuitos são:

$$f = m\dot{v} + \beta_{lin}v \qquad \tau = J\dot{\omega} + \beta_{ang}\omega$$

sendo m a massa total do minirobô, J o seu momento de inércia, β_{lin} é o coeficiente de atrito entre o robô e o solo durante os movimentos lineares e β_{ang} é o seu respectivo, para movimentos angulares. Na forma matricial tem-se:

$$\mathbf{f} = M_r \dot{\mathbf{v}} + B_r \mathbf{v} \tag{6.5}$$

onde:

$$\mathbf{f} = \begin{bmatrix} f \\ \tau \end{bmatrix} \qquad M_r = \begin{bmatrix} m & 0 \\ 0 & J \end{bmatrix} \qquad B_r = \begin{bmatrix} \beta_{lin} & 0 \\ 0 & \beta_{ang} \end{bmatrix} \qquad \dot{\mathbf{v}} = \begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix}$$

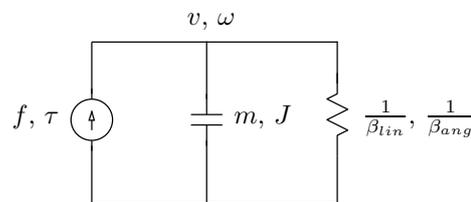


Figura 6.2: Análogo elétrico dos comportamentos linear e angular do robô

As forças exercidas sobre as rodas são $f_d = \tau_d/r_d$ e $f_e = \tau_e/r_e$ enquanto a força e o torque exercidos sobre o minirobô são:

$$f = f_d + f_e \qquad \tau = f_d \frac{d}{2} - f_e \frac{d}{2} \tag{6.6}$$

Substituindo f_d e f_e nas equações 6.6, têm-se:

$$f = \frac{1}{r_d}\tau_d + \frac{1}{r_e}\tau_e \qquad \tau = \frac{d}{2r_d}\tau_d - \frac{d}{2r_e}\tau_e$$

que, escrevendo na forma matricial fica:

$$\mathbf{f} = {}^\omega T_v^T \boldsymbol{\tau} \quad (6.7)$$

onde $\boldsymbol{\tau} = [\tau_d \ \tau_e]^T$.

Assim, igualando-se as equações 6.5 e 6.7, obtém-se:

$${}^\omega T_v^T \boldsymbol{\tau} = M_r \dot{\mathbf{v}} + B_r \mathbf{v} \quad (6.8)$$

Com relação à dinâmica dos atuadores, os análogos elétricos dos motores direito e esquerdo são apresentados na figura 6.3. As nomenclaturas adotadas são:

- $e_{d,e}$ é a tensão de armadura dos motores direito e esquerdo, respectivamente;
- $1/\rho_{d,e}$ é a resistência do enrolamento de armadura dos motores;
- $i_{d,e}$ é a corrente de armadura dos motores;
- $K_{d,e}$ é a constante de torque dos motores;
- $J_{d,e}$ é o momento de inércia combinado do rotor dos motores e das rodas;
- $\beta_{d,e}$ é o coeficiente de atrito viscoso angular entre os rotores e eixos dos motores e seus apoios.

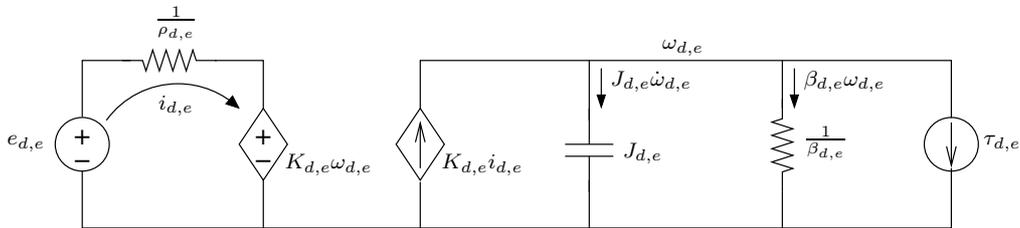


Figura 6.3: Análogo elétrico dos motores direito e esquerdo do minirobô

Deste circuito é possível obter as seguintes equações:

$$\mathbf{i} = R_m \mathbf{e} - R_m K_m \boldsymbol{\omega} \quad (6.9)$$

$$\boldsymbol{\tau} = K_m \mathbf{i} - J_m \dot{\boldsymbol{\omega}} - B_m \boldsymbol{\omega} \quad (6.10)$$

onde:

$$\mathbf{i} = \begin{bmatrix} i_d \\ i_e \end{bmatrix} \qquad \mathbf{e} = \begin{bmatrix} e_d \\ e_e \end{bmatrix} \qquad \dot{\boldsymbol{\omega}} = \begin{bmatrix} \dot{\omega}_d \\ \dot{\omega}_e \end{bmatrix} \qquad R_m = \begin{bmatrix} \rho_d & 0 \\ 0 & \rho_e \end{bmatrix}$$

$$K_m = \begin{bmatrix} K_d & 0 \\ 0 & K_e \end{bmatrix} \qquad J_m = \begin{bmatrix} J_d & 0 \\ 0 & J_e \end{bmatrix} \qquad B_m = \begin{bmatrix} \beta_d & 0 \\ 0 & \beta_e \end{bmatrix}$$

Substituindo a equação 6.9 na equação 6.10, tem-se:

$$\boldsymbol{\tau} = R_m K_m \mathbf{e} - J_m \dot{\boldsymbol{\omega}} - (R_m K_m^2 + B_m) \boldsymbol{\omega}$$

Da equação 6.1, tem-se que a derivada de $\boldsymbol{\omega}$ em relação ao tempo é dada por $\dot{\boldsymbol{\omega}} = {}^\omega T_v \dot{\mathbf{v}}$. Assim:

$$\boldsymbol{\tau} = R_m K_m \mathbf{e} - J_m {}^\omega T_v \dot{\mathbf{v}} - (R_m K_m^2 + B_m) {}^\omega T_v \mathbf{v}$$

que multiplicando por ${}^\omega T_v^T$ fica:

$${}^\omega T_v^T \boldsymbol{\tau} = ({}^\omega T_v^T R_m K_m) \mathbf{e} - ({}^\omega T_v^T J_m {}^\omega T_v) \dot{\mathbf{v}} - [{}^\omega T_v^T (R_m K_m^2 + B_m) {}^\omega T_v] \mathbf{v} \quad (6.11)$$

Igualando-se as equações 6.8 e 6.11 e isolando o termo em \mathbf{e} , obtém-se:

$${}^\omega T_v^T R_m K_m \mathbf{e} = (M_r + {}^\omega T_v^T J_m {}^\omega T_v) \dot{\mathbf{v}} + [B_r + {}^\omega T_v^T (R_m K_m^2 + B_m) {}^\omega T_v] \mathbf{v}$$

que representa o modelo matemático do robô no referencial dos eixos principais x e y . Este modelo pode ser expresso em forma matricial como:

$$K_v \mathbf{e} = M_v \dot{\mathbf{v}} + B_v \mathbf{v} \quad (6.12)$$

onde K_v é um ganho das entradas de controle e M_v e B_v são, respectivamente, a matriz de inércia e a matriz de atritos viscosos, ambas constantes, simétricas, definidas positivas. Estas três matrizes são dadas por:

$$\begin{aligned} K_v &= {}^\omega T_v^T R_m K_m \\ M_v &= M_r + {}^\omega T_v^T J_m {}^\omega T_v \\ B_v &= B_r + {}^\omega T_v^T (R_m K_m^2 + B_m) {}^\omega T_v \end{aligned}$$

e os seus elementos são:

$$\begin{aligned} M_v &= \begin{bmatrix} m + \frac{J_d}{r_d^2} + \frac{J_e}{r_e^2} & \frac{d}{2} \left(\frac{J_d}{r_d^2} - \frac{J_e}{r_e^2} \right) \\ \frac{d}{2} \left(\frac{J_d}{r_d^2} - \frac{J_e}{r_e^2} \right) & J + \frac{d^2}{4} \left(\frac{J_d}{r_d^2} + \frac{J_e}{r_e^2} \right) \end{bmatrix} & K_v &= \begin{bmatrix} \frac{\rho_d K_d}{r_d} & \frac{\rho_e K_e}{r_e} \\ \frac{d \rho_d K_d}{2 r_d} & -\frac{d \rho_e K_e}{2 r_e} \end{bmatrix} \\ B_v &= \begin{bmatrix} \beta_{lin} + \frac{\beta_d + \rho_d K_d^2}{r_d^2} + \frac{\beta_e + \rho_e K_e^2}{r_e^2} & \frac{d}{2} \left(\frac{\beta_d + \rho_d K_d^2}{r_d^2} - \frac{\beta_e + \rho_e K_e^2}{r_e^2} \right) \\ \frac{d}{2} \left(\frac{\beta_d + \rho_d K_d^2}{r_d^2} - \frac{\beta_e + \rho_e K_e^2}{r_e^2} \right) & \frac{d^2}{4} \left(\frac{\beta_d + \rho_d K_d^2}{r_d^2} + \frac{\beta_e + \rho_e K_e^2}{r_e^2} \right) \end{bmatrix} \end{aligned}$$

Além deste modelo no referencial dos atuadores, pode-se também derivar um modelo em variáveis de estado. Definindo-se os vetores:

$$\mathbf{x} = \begin{bmatrix} \mathbf{v} \\ \dots \\ \mathbf{q} \end{bmatrix} = \begin{bmatrix} v \\ \omega \\ x \\ y \\ \theta \end{bmatrix} \quad \dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{v}} \\ \dots \\ \dot{\mathbf{q}} \end{bmatrix}$$

pode-se obter um modelo do robô no espaço de estados a partir das equações 6.4 e 6.12:

$$\begin{cases} \dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{e} \\ \mathbf{y} = C\mathbf{x} \end{cases}$$

onde:

$$A(\theta) = \begin{bmatrix} -M_v^{-1} B_v & \vdots & 0 \\ \dots & \dots & \dots \\ {}^q T_v & \vdots & 0 \end{bmatrix} \quad B = \begin{bmatrix} M_v^{-1} K_v \\ \dots \\ 0 \end{bmatrix}$$

A matriz C depende de quais sinais são relevantes na aplicação onde o robô vai ser empregado. Como normalmente se deseja controlar a posição e a orientação do robô:

$$\mathbf{y} = \mathbf{q} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \rightarrow C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

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. Chegou-se ao seguinte conjunto de valores, com todas as unidades no S.I.:

$$\begin{aligned} d &= 0.0750 & r_e &= 0.0254 \\ r_d &= 0.0251 & J &= 13.02 \times 10^{-6} \\ m &= 0.352 & \beta_{ang} &= 7.004 \times 10^{-5} \\ \beta_{lin} &= 0.1002 & J_e &= 3.005 \times 10^{-6} \\ J_d &= 3.003 \times 10^{-6} & \beta_e &= 7.002 \times 10^{-6} \\ \beta_d &= 7.002 \times 10^{-6} & K_e &= 2.01 \times 10^{-2} \\ K_d &= 2.04 \times 10^{-2} & \rho_e &= 0.0578 \\ \rho_d &= 0.0577 \end{aligned}$$

Utilizando estes valores, chega-se a um modelo cujo comportamento em malha aberta e com entradas iguais é apresentado pela figura 6.6. As próximas seções abordarão a estratégia de controle adotada e o simulador do minirobô.

6.2 Controle dos Minirobôs

Como foi visto no capítulo 3, controladores lineares clássicos não estabilizam um sistema não-holonômico. Entretanto, segundo Pereira [PCA00], é possível utilizar controladores lineares simples caso o robô seja controlado por partes. Pelo modelo do robô apresentado na figura 6.1, sabe-se que têm-se duas variáveis de controle, que são as tensões nos atuadores direito e esquerdo para controlar três variáveis de saída, que formam a configuração do robô. As variáveis de controle no referencial de atuadores são não-triviais para serem trabalhadas devido ao acoplamento existente entre elas. Porém, se for considerado que o robô possui duas malhas de controle distintas, uma para movimentos puramente lineares, com variável de controle v , e outra para movimentos rotacionais, com variável ω , é possível obter um desacoplamento nas variáveis de controle e, dessa forma, obter malhas distintas. Os dois controladores são chamados de linear e angular, respectivamente.

6.2.1 Estratégia de Controle

O movimento completo do robô é composto de movimentos lineares e rotacionais conjugados. Porém, o controle do movimento linear somente tem sentido caso o robô esteja com a orientação desejada para alcançar o alvo, ou seja, quando a variável de configuração θ estiver sendo controlada.

Para obter esta estratégia faz-se o robô aproximar-se de uma posição provisória (x_p, y_p) situada a uma distância Δl da posição de referência final (x_f, y_f) , como exemplificado

na figura 6.4. Esta posição provisória situa-se na reta que passa pelo ponto final com inclinação igual à $\tan \theta_f$. Esta distância Δl é proporcional à menor distância entre o robô e a reta com inclinação $\tan \theta_f$ que passa por (x_f, y_f) através da seguinte relação:

$$\Delta l = 2r_{med} \cdot (1 - e^{-\ln 2d/r_{med}})$$

onde r_{med} é uma distância de recuo médio cujo valor escolhido foi 0,1m.

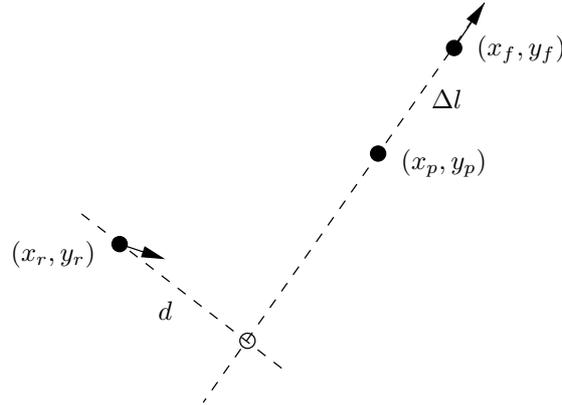


Figura 6.4: Estratégia do controle para aproximação da posição de referência final

Este artifício tem o objetivo de fazer com que a posição de aproximação do robô coincida com a orientação final desejada, o que minimiza a necessidade da manobra final de alinhamento do robô com θ_f . Isto é possível graças ao movimento que o robô faz em direção a este alvo provisório, uma vez que ele tende a aproximar-se cada vez mais da reta que passa por (x_f, y_f) com inclinação $\tan \theta_f$, a distância Δl tende a igualar-se a zero. O erro do controlador linear será a distância euclidiana entre (x_r, y_r) e (x_p, y_p) multiplicado pelo cosseno do erro angular. Este produto é necessário para garantir que o robô somente deva se mover quando o controlador angular iniciar o ajuste da orientação do robô para a direção do alvo.

O erro angular (equação 6.13) é simplesmente a diferença entre a orientação atual do robô e a direção de referência do movimento, caso o robô esteja afastado do alvo, ou é a própria orientação final, caso o robô esteja suficientemente próximo da posição de referência. Neste caso, o controlador linear deixa de atuar, uma vez que o erro linear é aceitável, e o controlador angular passa a ajustar a orientação do robô.

$$\theta_{ref} = \begin{cases} \theta_f, & \text{se o robô estiver próximo do alvo;} \\ \tan^{-1} \left(\frac{y_p - y_r}{x_p - x_r} \right), & \text{caso contrário.} \end{cases} \quad (6.13)$$

Os dois controladores utilizados são do tipo PID cujos ganhos foram obtidos empiricamente através da observação do desempenho dos controladores. Seus valores são:

$$\begin{array}{ll} K_p^{lin} = 1,0 & K_p^{ang} = 0,1 \\ K_i^{lin} = 0,1 & K_i^{ang} = 0,01 \\ K_d^{lin} = 0,01 & K_d^{ang} = 0,001 \end{array}$$

As duas variáveis de controle v e ω são calculadas de forma independente porém o movimento final do robô, composto de movimentos lineares e angulares, é resultado da

ponderação destas duas variáveis sobre as tensões dos motores direito e esquerdo. Ambos os sinais de controle são então normalizados entre o intervalo $[+1, -1]$, sendo que $+1$ corresponde às máximas velocidades positivas e -1 , às mínimas negativas. Uma vez tendo os sinais v e ω normalizados, calcula-se a tensão dos motores através de:

$$e_d = (v + \omega) \cdot V_{cc}$$

$$e_e = (v - \omega) \cdot V_{cc}$$

onde V_{cc} é a tensão nominal, cujo valor é $9V$, aplicada ao motor.

Esta estratégia de controle é adequada para um sistema de rastreamento de trajetória, onde um robô deve seguir um alvo móvel por exemplo. Quando aplicado para o problema de estabilização, ou seja, quando o robô deve alcançar uma configuração fixa no espaço de trabalho, os resultados obtidos não são satisfatórios, como observado nos exemplos da figura 6.5. Entretanto, estes controladores devem ser utilizados em conjunto com o planejador de caminho e o gerador de trajetória, o que resulta em um controle mais preciso para o minirobô.

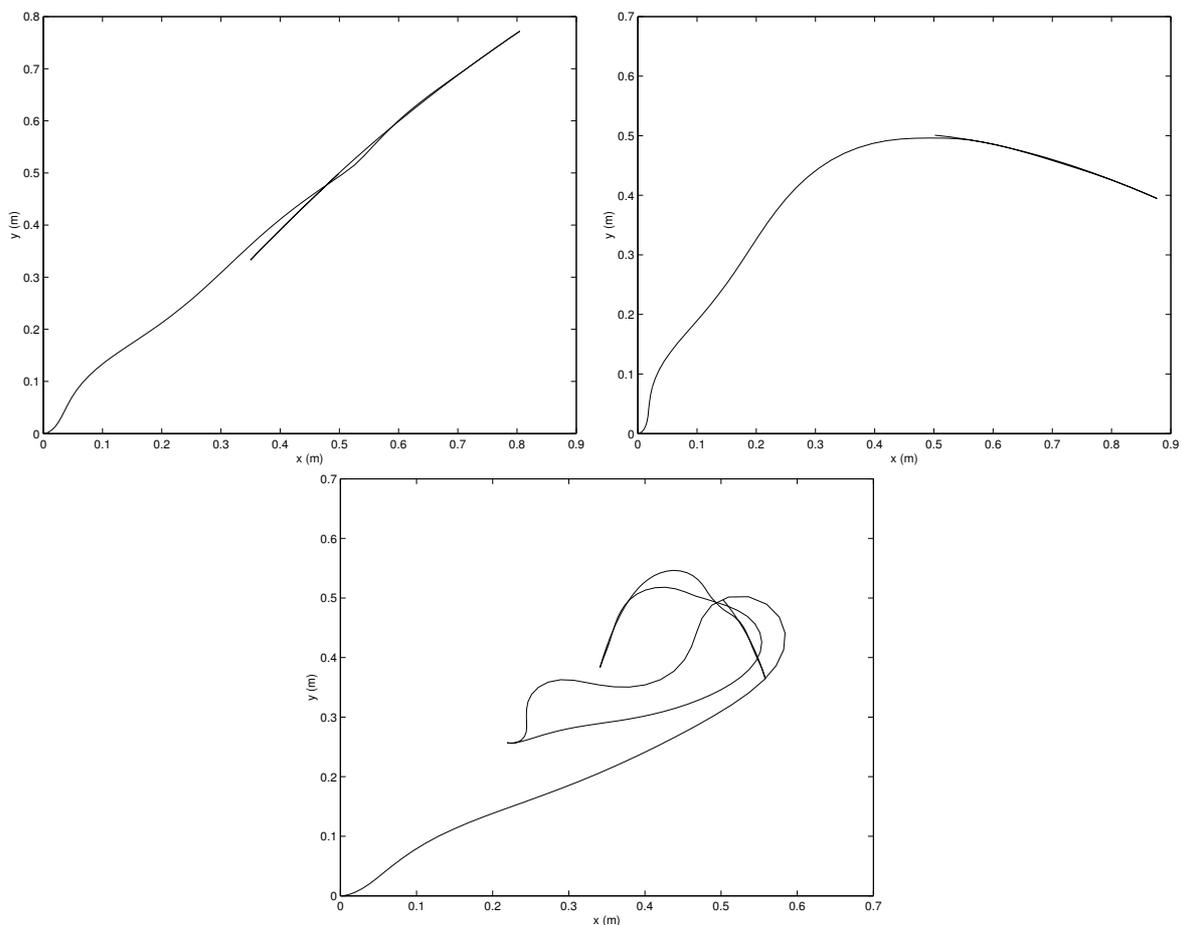


Figura 6.5: Exemplos de estabilização do minirobô em uma configuração fixa

6.3 Simulador dos Minirroboês

Para verificar a validação dos métodos de planejamento de caminhos e geração de trajetórias foi implementado um simulador dos minirroboês baseado no modelo dinâmico encontrado. Este simulador apresenta-se de maneira bastante simples, consistindo apenas na atualização do vetor de estados do robô, sem a necessidade de reproduzir o ambiente do futebol. Neste simulador apenas um robô foi modelado e, portanto, não há a preocupação com colisões com obstáculos.

Utilizando as equações e os valores dos parâmetros do minirroboê apresentados na seção 6.1, encontra-se:

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -0,5436 & -1,379 \cdot 10^{-4} \\ -3,027 & -7,819 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} + \begin{bmatrix} 0,129 & 0,127 \\ 66,915 & -65,304 \end{bmatrix} \begin{bmatrix} e_d \\ e_e \end{bmatrix}$$

Como exemplo do desempenho do simulador, a figura 6.6 apresenta o comportamento do minirroboê submetido a entradas iguais nos dois motores ($e_d = 7,2V$ e $e_e = 7,2V$) e sem controle de execução da trajetória, ou seja, em malha aberta. Pelo comportamento apresentado, percebe-se que o robô simulado pode perfeitamente servir como plataforma de testes para o planejamento de caminho, geração e execução de trajetória, uma vez que o movimento reproduzido por este robô simulado é similar ao movimento do robô real sob as mesmas condições.

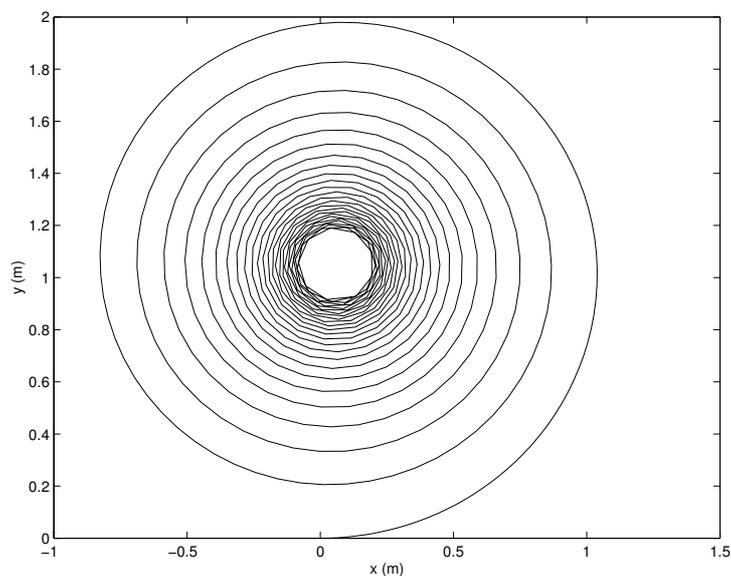


Figura 6.6: Comportamento em malha aberta do minirroboê simulado, submetido a entradas iguais

O programa do simulador foi escrito em linguagem C++ e consiste em, dado uma configuração de referência, gerar sucessivas trajetórias entre a atual configuração do minirroboê e a configuração final, baseado no método de geração de trajetória apresentado no capítulo anterior, e controlar a execução desta trajetória através de um sistema de controle simples apresentado na seção 6.2. Como a trajetória é continuamente recalculada, o caminho final percorrido pelo robô difere do caminho inicial que é calculado pelo método dos polinômios cúbicos. A próxima seção apresentará alguns resultados obtidos.

6.4 Resultados Experimentais

Os resultados obtidos com o simulador foram úteis para validar o funcionamento do planejamento de caminho e geração de trajetória a cada intervalo de amostragem do sistema, além de demonstrar que um controlador simples capaz de rastrear uma trajetória trabalhando em conjunto com um gerador de referências pode perfeitamente ser aplicado para robôs móveis com acionamento diferencial, como apresentado no capítulo 3. O modelo do robô utilizado foi deduzido na seção 6.1. Ele possui a vantagem de incluir a dinâmica dos atuadores, o que não é realizado em outros modelos encontrados na literatura. A figura 6.6 mostra comportamento em malha aberta do robô submetido a duas entradas iguais nos seus motores. Percebe-se que devido a assimetrias em sua estrutura, ou seja, diferenças entre motores, distribuição irregular de massa, etc., o minirobô simulado desenvolve movimentos espirais no espaço de trabalho.

Ao incluir as etapas de planejamento de caminho, geração e controle de trajetória, obteve-se um comportamento mais adequado para minirobô, o que pode ser visualizado nos exemplos da figura 6.7. Nestes exemplos foi adotado como configuração inicial $(0,0, 0,0, 0,0)$ e como configuração final foi escolhido o ponto $(0,5, 0,5)$ e três ângulos finais, cujos valores foram $5\pi/18$, 0 e $-5\pi/18$. A linha tracejada indica o caminho original entre as duas configurações e a linha contínua mostra o movimento final do minirobô sob a ação completa do controle.

Nestes exemplos também pode ser observado que o percurso executado difere do caminho original. Isto pode ser explicado pelo fato de que um novo caminho sempre é calculado entre a configuração atual do robô e a configuração final desejada, porém ressalta-se que o robô sempre alcança esta configuração final com erro próximo de zero. O fato dos movimentos finais diferirem dos caminhos iniciais calculados não invalida os métodos propostos neste trabalho, pois estes movimentos finais também respeitam a restrição não-holonômica do minirobô. Estes resultados apresentam-se satisfatórios, garantindo assim a eficácia dos métodos empregados.

6.5 Conclusões

Este capítulo apresentou os resultados experimentais do planejamento de caminho e geração de trajetórias obtidos a partir de testes realizados em um simulador do minirobô. Viu-se que o modelo matemático do robô apresenta-se de forma mais completa possível pois inclui a dinâmica dos atuadores. Não foram apresentadas provas de estabilidade do controlador projetado pois buscou-se um projeto simples, como o controlador proposto por Pereira [PCA00], capaz de seguir uma referência móvel. Era esperado que, ao utilizar as etapas de planejamento de caminho e geração de trajetória juntamente com o controle, o robô alcançasse a configuração final. Isto foi apresentado nos exemplos da figura 6.7. Apesar dos resultados serem obtidos de um simulador, eles foram fidedignos o suficiente de um sistema real, o que permite a sua utilização em uma plataforma real. Percebeu-se também que, sendo as trajetórias calculadas a cada intervalo de amostragem, o percurso final realizado pelo robô foi constituído de um conjunto de caminhos obtidos através dos polinômios cúbicos, diferindo do caminho original entre as configurações inicial e final. Esta diferença não influencia a aplicabilidade dos métodos propostos pois verificou-se que os movimentos executados pelo minirobô respeitaram a restrição não-holonômica.

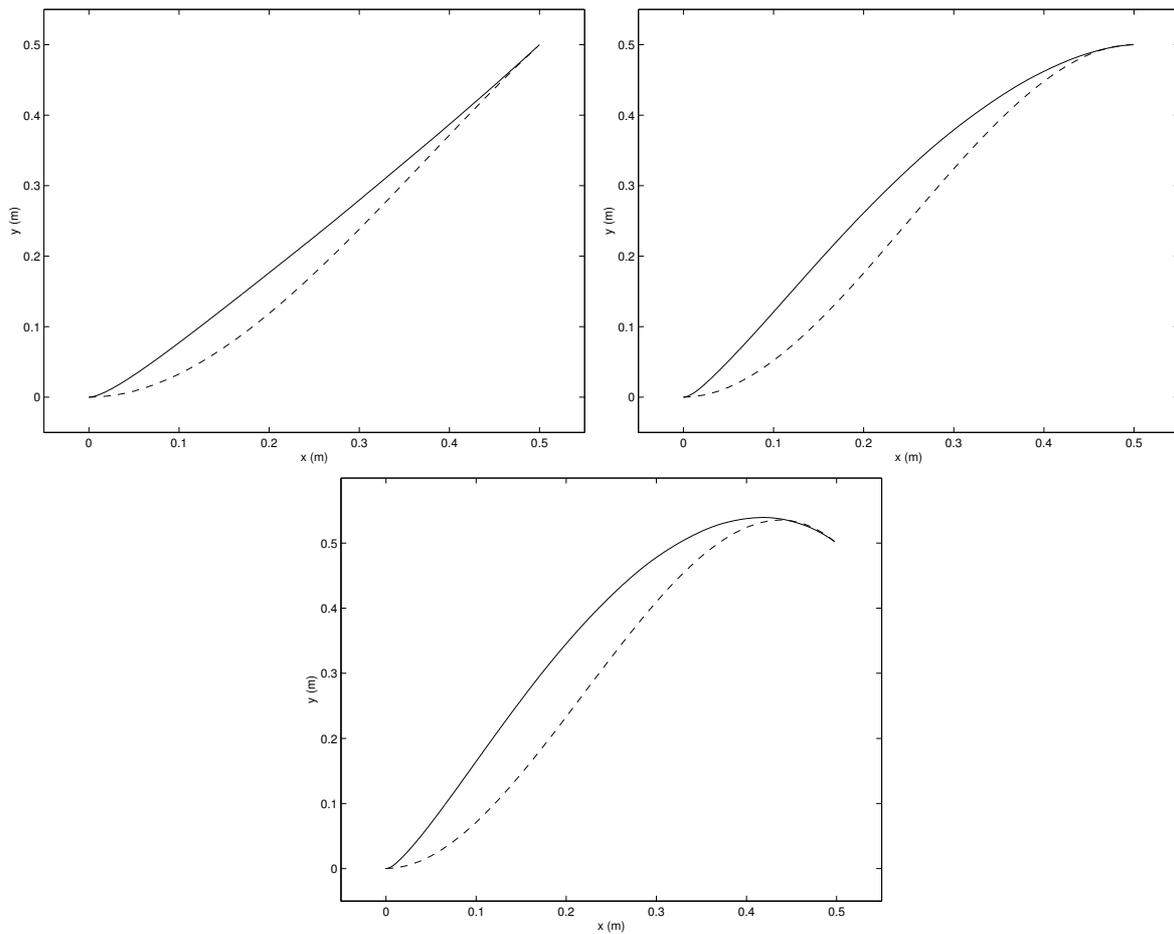


Figura 6.7: Comportamento do minirobô quando são utilizados o planejamento de caminho, geração e controle de trajetória

O próximo capítulo apresentará as conclusões finais e os possíveis futuros trabalhos que podem ser implementados de forma contribuir com este sistema de navegação.

Capítulo 7

Conclusões Finais e Perspectivas Futuras

Neste trabalho foi apresentado uma proposta para navegação de robôs móveis autônomos cuja atuação se dá em um ambiente dinâmico. A navegação é a principal tarefa que pode ser atribuída a um robô móvel. Para executá-la eficientemente, ele precisa localizar-se em seu ambiente de trabalho, planejar um caminho adequado, que o ligue a um alvo especificado, gerar uma trajetória adicionando restrições de tempo ao caminho encontrado e determinando perfis de velocidade do robô, e executar esta trajetória no ambiente de trabalho. A proposta inicial foi aplicar os métodos desenvolvidos numa plataforma de futebol de minirobôs móveis, cuja descrição foi realizada no capítulo 2. Os robôs móveis do projeto do futebol possuem uma particularidade que se expressa nos seus movimentos. Devido ao fato de possuírem rodas que impedem a sua movimentação lateral, eles são denominados sistemas com restrição não-holonômica. Tal restrição permite que os minirobôs movimentem-se apenas para onde apontam o seus vetores velocidade. Esta característica física do robô somada ao ambiente dinâmico do futebol, como o constante movimento de robôs e colisões, aumenta a dificuldade para determinar um caminho que possa ser percorrido. Uma outra restrição imposta ao desenvolvimento de uma plataforma de futebol é que todas as ações relativas ao comportamento do robô, tal como processamento do sistema de visão, estratégias de movimento, controle, entre outros, fossem realizadas durante os intervalos de amostragem do sistema de visão do projeto, cuja taxa é de $33ms$.

Este trabalho não tratou da localização dos minirobôs no campo de futebol, através da análise de imagem. Isto foi tratado por Aires [Air01] em seu trabalho de mestrado. Coube a este trabalho apresentar uma ferramenta auxiliar que permitisse que, durante todo o tempo de execução do jogo, os módulos de planejamento de caminho, geração de trajetória e controle do robô tivessem acesso às variáveis de configuração dos robôs. Foi explicado que sob determinadas condições, as configurações dos minirobôs não são calculadas. Para evitar que esta ausência de dados implicasse em erros nos outros módulos do projeto de futebol, implementou-se um sistema que estima as variáveis de configuração do robô baseado no algoritmo dos Mínimos Quadrados. As variáveis de configuração do minirobô foram linearizadas para que o algoritmo pudesse ser executado corretamente. Os resultados do estimador foram apresentados no capítulo 4. As aquisições de dados foram feitas com um robô movendo-se de forma aleatória, e em malha aberta, com a estimação realizada *a posteriori*. Os resultados obtidos foram satisfatórios, com reduzidos

erros absolutos para x e y . Os resultados para a variável de configuração θ precisaram de um tratamento auxiliar pois foi necessário normalizar o ângulo θ entre $(-\infty, \infty)$ para tornar a sua representação linear e capaz de ser estimada. Isto permitiu a aplicabilidade do algoritmo dos Mínimos Quadrados e, com isso, o desenvolvimento completo de uma ferramenta de estimação da configuração dos robôs. Entretanto, ressalta-se que a sua utilização ocorre somente quando os minirobôs não são localizados pelo sistema de visão.

O capítulo 5 apresentou a maior contribuição deste trabalho para a solução do problema da navegação em robôs móveis autônomos. A idéia foi criar um caminho capaz de ser percorrido através da utilização de polinômios paramétricos de terceiro grau em x e y . Estes polinômios possuem coeficientes que podem ser recalculados a cada passo de amostragem cujo desenvolvimento foi apresentado no apêndice B. Os caminhos encontrados se mostraram adequados para os minirobôs porque o ângulo θ foi imposto de forma a satisfazer à restrição não-holonômica. Polinômios de terceiro grau foram escolhidos para determinar a configuração do robô porque eles possuem graus de liberdade suficiente para ajustar o caminho de forma a evitar que o robô execute movimentos desnecessários. Uma vez determinado o caminho, foi necessário atribuir perfis de velocidade para o robô, gerando a sua trajetória. A idéia fundamental para a geração de trajetória foi que o minirobô percorresse o caminho com uma velocidade linear constante. Isto é fisicamente irrealizável devido às não-linearidades do sistema, tal como colisões com obstáculos, atrasos entre o envio do sinal de controle e a resposta dos atuadores do minirobô, entre outros. Entretanto, o método desenvolvido permite que, computacionalmente, o caminho seja percorrido por completo considerando uma velocidade constante. Isto ocorre através do cálculo do parâmetro λ que é necessário para que o robô, no próximo intervalo de amostragem, esteja a uma distância pré-determinada de sua posição atual. Os resultados apresentados no capítulo 5 mostraram três exemplos de trajetórias onde as configurações do minirobô são, de forma aproximada, uniformemente espaçadas, o que atende ao objetivo determinado.

Este trabalho também apresentou uma modelagem matemática completa do minirobô, pois incluiu a dinâmica dos atuadores, que tornou-se útil para a elaboração de uma plataforma para testes. Este simulador serviu para testar a aplicabilidade dos métodos de planejamento de caminho, geração de trajetória e para a própria validação do modelo matemático do robô. Para garantir que o minirobô simulado executasse os movimentos necessários para alcançar o alvo, foi elaborada uma estratégia simples de controle, apresentada no capítulo 6. Esta estratégia baseou-se no desmembramento do robô em duas malhas de controle distintas, sendo uma para movimentos puramente lineares, e outra para movimentos puramente rotacionais. Porém o sistema de controle completo consistiu nas etapas sequenciais de planejamento de caminho, geração de trajetória e controle do robô ou, de forma simplificada, em um gerador de referências e no controle propriamente dito do robô, como proposto no capítulo 3. Os resultados apresentados no capítulo 6 mostraram que o planejamento de caminho e geração de trajetória cumpriram os objetivos propostos no capítulo 1. Contudo, percebeu-se que o percurso final desenvolvido pelo robô difere do caminho original entre as configurações inicial e final. Isto é explicado pelo fato do caminho ser recalculado a cada nova posição do minirobô, resultando assim em outros novos caminhos que são distintos do primeiro. Entretanto, isto não prejudicou o método, uma vez que o controle completo (gerador de referência mais controle do robô) permitiram que o minirobô desenvolvesse um movimento contínuo, isto é, sem a necessidade de parar para ajustar a sua orientação, e respeitando a restrição não-holonômica.

O fato do caminho poder ser recalculado a cada intervalo de amostragem é útil porque permite ao robô perseguir, por exemplo, a bola que está em constante movimento.

Este trabalho também permite a possibilidade de se obter várias extensões. Com relação à estimação do minirobô, baseada apenas nos dois valores imediatamente passados de sua configuração e de suas entradas, permite que se possa prever, dentro de um limite de erro aceitável, onde jogadores adversários estarão posicionados e, desta maneira, modificar estratégias de jogo. Isto será bastante útil para o sistema gerenciador da equipe de futebol. Outros métodos de estimação também podem ser utilizados para que se possa fazer um teste comparativo e aplicar o mais adequado para o sistema proposto.

Para o método de geração de caminhos utilizando polinômios de terceiro grau pode-se verificar novos métodos para a sua adequação como, por exemplo, o menor caminho que respeite a restrição não-holonômica, caminho que possa ser percorrido em menor tempo, ou então uma junção destes métodos isolados. Pode-se determinar algum critério para a escolha de qual direção x ou y deverá ter o caminho reduzido. Isto é particularmente importante nos casos em que não é possível encontrar os valores para os coeficientes livres de forma a satisfazer os critérios de adequação impostos. Deve-se verificar outros métodos para encontrar os caminhos como polinômios de grau superior, curvas de Bezier, *splines*, clotóides e realizar um teste comparativo entre eles. Como tais caminhos são apropriados para robôs com acionamento diferencial, pode-se utilizar técnicas com a mesma idéia básica para robôs com *trailers* e robôs semelhantes a carros, o que permitira obter uma ferramenta cuja aplicabilidade abrangeria diversos tipos de robôs. Outros métodos para a geração de trajetória podem ser elaborados. Uma proposta interessante é criar uma maneira de gerar as referências de forma que leve em consideração o tempo necessário para alcançar a velocidade máxima que pode ser executada pelo robô, e o tempo total para ele parar completamente, após cessados os sinais de controle.

Embora o modelo matemático do minirobô seja completo, do ponto de vista de inclusão da dinâmica dos atuadores, os valores de seus parâmetros necessitam ser determinados de forma mais rigorosa. Há duas propostas em questão: ensaios de laboratório para determinação de valores considerados difíceis de serem medidos, como momento de inércia, coeficiente de atrito das rodas com o solo, dos mancais dos motores, da carcaça do robô com o solo, etc., ou então utilizar recursos de identificação computacional de parâmetros. Este último apresenta-se mais adequado por duas razões principais. A primeira é que pode-se obter valores bastante apropriados para o modelo em espaço de estado. A segunda razão é que podem ser utilizados durante a execução do jogo de futebol, o que permitiria atualizar constantemente os parâmetros variantes no tempo. Uma vez determinado o modelo exato, ou tão próximo quanto possível, do robô pode-se elaborar outras estratégias de controle, sobretudo controladores para sistemas não-lineares, multivariáveis e invariantes no tempo, além da exigência da prova de estabilidade das leis de controle propostas.

Apesar dos aspectos não previstos que ocorreram, este trabalho trouxe inovações científicas para a resolução da problemas ligados à navegação de robôs móveis autônomos com acionamento diferencial. Espera-se que esta contribuição seja útil para outros projetos de sistemas robóticos, deste e de outros centros de pesquisa.

Referências Bibliográficas

- [AA99] Rui Araújo and Aníbal T. de Almeida. Learning sensor-based navigation of a real mobile robot in unknown worlds. *IEEE Transactions on Systems, Man, and Cybernetics*, 29(2), April 1999. Part B: Cybernetics.
- [Agu00] Luis Antônio Aguirre. *Introdução à Identificação de Sistemas: Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. Editora UFMG, 2000.
- [Air01] Kelson Rômulo Teixeira Aires. Desenvolvimento de um sistema de visão global para uma frota de mini-robôs móveis. Dissertação de Mestrado, PPGEE-UFRN, Natal, RN, Brasil, março 2001.
- [AK99] G. C. Anousaki and K. J. Kyriakopoulos. Simultaneous localization and map building for mobile robot navigation. *IEEE Robotics and Automation Magazine*, September 1999.
- [ÅW97] Karl J. Åström and Björn Wittenmark. *Computer-Controlled Systems: Theory and Design*. Thomas Kailath, series editor, 1997. Third edition.
- [Bor98] Geovany Araújo Borges. Um sistema óptico de reconhecimento de trajetórias para veículos automáticos. Dissertação de Mestrado, COPELE-CCT-UFPB, Campina Grande, PB, Brasil, junho 1998.
- [CA98] Jen-Hui Chuang and Narendra Ahuja. An analytically tractable potential field model of free space and its application in obstacle avoidance. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(5), October 1998. Part B: Cybernetics.
- [CB95a] Howie Choset and Joel Burdick. Sensor based planning, part 01: The generalized voronoi graph. In *Proceedings of the IEEE International Conference of Robotics and Automation*, Nagoya, Japan, 1995.
- [CB95b] Howie Choset and Joel Burdick. Sensor based planning, part 02: Incremental construction of the generalized voronoi graph. In *Proceedings of the IEEE International Conference of Robotics and Automation*, Nagoya, Japan, 1995.
- [Cha95] Raja Chatila. Deliberation and reactivity in autonomous mobile robot. *Robotics and Autonomous Systems*, 16, 1995.
- [CP00] Anna Helena Reali Costa e René Pegoraro. Construindo robôs autônomos para partidas de futebol: O time guaraná. *SBA Controle e Automação*, 11(3), 2000.

- [CPS⁺99] A. H. Reali Costa, R. Pegoraro, G. Stolfi, J. S. Sichman, F. M. Pait, e H. Ferasoli Filho. Guaraná robot-soccer team: Some architectural issues. In *Anais do IV Simpósio Brasileiro de Automação Inteligente*, São Paulo, SP, Brasil, setembro 1999.
- [Cra86] John J. Craig. *Introduction to Robotics: Mechanics and Control*. Addison-Wesley Publishing Company, 1986.
- [CZ97] M. W. Chen and A. M. S. Zalzalá. Dynamic modelling and genetic-based trajectory generation for non-holonomic mobile manipulators. *Control Engineering Practice*, 05(01), 1997.
- [DW97] Adam W. Divelbiss and John T. Wen. A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Transactions on Robotics and Automation*, 13(3), June 1997.
- [EG98] Ashraf Elnagar and Kamal Gupta. Motion prediction of moving objects based on autoregressive model. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(6), November 1998. Part A: Systems and Humans.
- [Fer98] Pierre Ferbach. A method of progressive constraints for nonholonomic motion planning. *IEEE Transactions on Robotics and Automation*, 14(1), February 1998.
- [FHB93] Guy Foux, Michel Heymann, and Alfred Bruckstein. Two-dimensional robot navigation among unknown stationary polygonal obstacles. *IEEE Transactions on Robotics and Automation*, 9(1), February 1993.
- [FNG97] Atsushi Fujimori, Peter N. Nikiforuk, and Madan M. Gupta. Adaptive navigation of mobile robots with obstacle avoidance. *IEEE Transactions on Robotics and Automation*, 13(4), August 1997.
- [HA92] Yong K. Hwang and Narendra Ahuja. A potential field approach to path planning. *IEEE Transactions on Robotics and Automation*, 8(1), February 1992.
- [HK99] Andrew Howard and Les Kitchen. Navigation using natural landmarks. *Robotics and Autonomous Systems*, 26, 1999.
- [HV98] Kwun Han and Manuela Veloso. Reactive visual control of multiple non-holonomic robotic agents. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.
- [KK99] Ashraf A. Kassim and B. V. K. Vijaya Kumar. Path planner based on the wave expansion neural network. *Robotics and Autonomous Systems*, 26, 1999.
- [KVV98] Jong-Hwan Kim, Prahlad Vadakkepat, and Igor M. Verner. The fira robot world cup initiative and research directions. Communicated to the International Journal of Robotics and Autonomous Systems, 1998.
- [Lat91] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Press, 1991.

- [Lau98] Jean-Paul Laumond. *Robot Motion Planning and Control*. CNRS-LAAS, 1998.
- [LH00] Walter Fetter Lages e Elder Moreira Hemerly. Controle de robôs móveis utilizando transformação descontínua e linearização adaptativa. In *Anais do XIII Congresso Brasileiro de Automática*, Florianópolis, SC, Brasil, setembro 2000.
- [LJTM94] Jean-Paul Laumond, Paul E. Jacobs, Michel Taïx, and Richard M. Murray. A motion planner for nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation*, 10(5), October 1994.
- [LS98] J. L. Díaz de León and J. H. Sossa. Automatic path planning for a mobile robot among obstacles of arbitrary shape. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(3), June 1998. Part B: Cybernetics.
- [Med98a] Adelardo Adelino Dantas de Medeiros. Introdução à robótica. In *XVII Encontro Nacional de Automática*, Natal, RN, Brasil, Julho 1998.
- [Med98b] Adelardo Adelino Dantas de Medeiros. A survey of control architectures for autonomous mobile robots. *Journal of the Brazilian Computer Society*, 4(3), April 1998. Special issue on robotics.
- [Oga97] Katsuhiko Ogata. *Modern Control Engeneering*. Pretence Hall, 1997. Third Edition.
- [OUV98] Giuseppe Oriolo, Giovanni Ulivi, and Marilena Vendittelli. Real-time map building and navigation for autonomous robots in unknown environments. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(3), June 1998. Part B: Cybernetics.
- [PCA00] Guilherme A. S. Pereira, Mário F. M. Campos, e Luis A. Aguirres. Modelo dinâmico para predição da posição e orientação de micro-robôs móveis observados por visão computacional. In *Anais do XIII Congresso Brasileiro de Automática*, Florianópolis, SC, Brasil, setembro 2000.
- [Per00] Guilherme Augusto Silva Pereira. Identificação e controle de micro-robôs móveis. Dissertação de Mestrado, CPDEE-UFMG, Belo Horizonte, MG, Brasil, fevereiro 2000.
- [RK92] Elon Rimon and Daniel E. Koditschek. Exact robot navigation using artificial potencial functions. *IEEE Transactions on Robotics and Automation*, 8(5), October 1992.
- [RN95] Stuart Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
- [RS90] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2), 1990.
- [SA00] Allan Aminadab A. Freire Soares e Pablo Javier Alsina. Mobile robot obstacle avoidance based on potencial field: A case study. In *Anais do Congresso Nacional de Engenharia Mecânica*, Natal, RN, Brasil, julho 2000.

- [SJH00] Celso Souza Jr. e Elder Moreira Hemerly. Controle de robôs móveis em coordenadas polares via rede neural. In *Anais do XIII Congresso Brasileiro de Automática*, Florianópolis, SC, Brasil, setembro 2000.
- [TF98] Roberto Fernandes Tavares Filho. Um breve histórico dos robôs no país do futebol, 1998.
- [VP99] Peter Veelaert and Herbert Peremans. Flexibility maps: A formalisation of navigational behaviors. *Robotics and Autonomous Systems*, 27, 1999.
- [YK98] Jung-Min Yang and Jong-Hwan Kim. Sliding mode motion control of nonholonomic mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

Apêndice A

Regras MiroSot

A.1 O Campo e a Bola

Dimensões do Campo

Consiste em um campo retangular preto (não-reflexivo) com dimensões $150\text{cm} \times 130\text{cm}$ e com paredes brancas de 5cm de altura com $2,5\text{cm}$ de espessura. Os topos das paredes são pintados de preto. São usados triângulos isóceles sólidos nos quatro vértices do campo (*corners*) para evitar que a bola fique presa. A textura da superfície do campo deve ser semelhante à textura de uma mesa de tênis-de-mesa.

Marcas no Campo

As marcações no campo são feitas como mostra a figura A.1. O círculo central tem 20cm de raio. O arco que faz parte da área do gol tem 20cm ao longo da linha do gol e 5cm perpendicular a ela. As maiores linhas e arcos são pintados de cor branca com 3mm de espessura. As posições do robô na bola livre são pintados de cor cinza.

O Gol

O gol tem 40cm de comprimento. Traves e redes não são usados.

A Linha do Gol e a Área do Gol

A linha do gol é uma linha de 40cm na frente do gol. A área do gol compreende um retângulo ($70\text{cm} \times 15\text{cm}$) e um arco de 20cm de comprimento ao longo do retângulo e com 5cm perpendicular a ele.

A Bola

Uma bola de golfe laranja com $42,7\text{mm}$ de diâmetro e 46g de massa.

Localização do Campo

O campo é usado em ambientes internos (*indoors*).

Condições de Iluminação

As condições de iluminação nas competições são fixadas em torno de $1.000Lux$.

A.2 Os Jogadores

O Sistema

Uma partida será jogada por dois times, cada um com três jogadores. Um dos robôs pode ser o goleiro. Três membros humanos do time (gerenciador, técnico e treinador) são permitidos na partida. Um computador por time, principalmente dedicado para processamento da visão e localização, pode ser usado.

Os Robôs

O tamanho de cada robô é limitado em um cubo de $7,5cm$ de lado. A altura da antena de comunicação RF não é considerada no tamanho do robô. O topo de um robô não deve ser colorido de laranja. Uma etiqueta colorida de azul ou amarelo, como decidido pelos organizadores identificará os robôs em um time. Todos os robôs devem ter ao menos uma região de $3,5cm \times 3,5cm$ de suas cores de etiqueta (azul ou amarelo) visível no topo. A identificação da cor do time mudará de jogo para jogo e a etiqueta de etiqueta deverá ser removível. Quando marcados com a etiqueta da cor de identificação, os robôs não podem ter quaisquer outras etiquetas nas cores do time oponente. Recomenda-se preparar um mínimo de seis etiquetas de cores diferentes de azul e amarelo para identificação individual. Para habilitar sensoriamento infravermelho os robôs podem ter luzes coloridas em seus lados exceto naqueles lugares usados essencialmente para funções do robô tal como lugares para sensores, rodas ou mecanismo de captura da bola. Eles podem usar uniformes de tamanho limitado a um cubo de $8cm$ de lado. Um robô em sua área de gol é considerado como goleiro. É permitido ao goleiro segurar a bola apenas quando ele estiver em sua área. Cada robô deve ser completamente independente, com potência e mecanismos motores próprios. Somente comunicação via rádio será permitida entre os jogadores e o *host* para envio de qualquer tipo de comando. É permitido que os robôs sejam equipados com braços, pernas, etc., mas eles devem respeitar as restrições de tamanho. Somente o goleiro, tem a permissão de capturar mais de 30% da bola, quando visto de um referencial externo. Enquanto o jogo estiver em progresso, em qualquer apito do juiz o operador humano pode parar todos os robôs usando a comunicação *host-robô*.

Substituições

Duas substituições são permitidas enquanto o jogo está em progresso. Elas serão realizadas mediante um pedido de tempo.

Pedido de Tempo

O operador humano pode fazer um pedido de parada de tempo de jogo. São permitidas duas paradas por time de dois minutos cada.

A.3 Transmissão de Informações

O gerenciador, o técnico ou o treinador podem transmitir certos comandos diretamente do computador remoto para seus robôs. Não é permitido transmitir sinais para parar um ou todos os robôs sem a permissão do juiz. Qualquer outra informação como estratégia de jogo, pode ser comunicada somente quando o jogo não estiver em progresso. O operador humano não pode controlar diretamente o robô com *joystick* ou teclado do computador sob quaisquer circunstâncias. Enquanto o jogo está em andamento, o *host* pode mandar qualquer informação autonomamente.

A.4 O Sistema de Visão

Com o intuito de identificar os robôs e a bola no campo, um sistema de visão deve ser usado. A localização da câmera do time será sobre a metade do campo do time. A localização da câmera deve ficar em uma altura igual ou superior a $2m$.

A.5 O Jogo

Duração

A duração do jogo é de dois períodos de tempo iguais a $5min$ cada, com um intervalo de tempo igual a $10min$. Após o intervalo, se algum dos times não estiver pronto para leitura de dados, um tempo adicional de $5min$ será dado para que sejam realizadas modificações corretivas. Caso o tempo se esgote e o time não estiver preparado, ele será desclassificado.

Início do Jogo

Antes do início do jogo, a cor dos times ou a posse de bola será decidida por moeda. Ao time que vencer, será permitido escolher a cor de identificação ou a bola. O time que escolher a bola pode optar pela frequência de transmissão. O time atacante pode ser colocado livremente na sua metade do campo. O time que se defende pode colocar seus jogadores em qualquer lugar de sua metade do campo, exceto na área circular. A bola deve ser colocada no centro do círculo. Com o apito do juiz, o jogo pode ser iniciado ou reiniciado e os robôs podem mover-se livremente. Após o intervalo, os times têm que mudar seus lados.

O Vencedor

Um gol é marcado quando a bola ultrapassa completamente a linha do gol. O vencedor do jogo é então escolhido com base no número de gols marcados pelos robôs.

A Prorrogação

Na ocorrência de um empate, o vencedor será decidido no esquema de morte súbita. Haverá uma prorrogação de $3min$ após um intervalo de $5min$. O time que marcar o primeiro gol será o vencedor. Caso o empate continue, o jogo será decidido através de pênaltis. Cada time terá direito a três chutes de pênaltis.

Faltas

Ocorre falta nos seguintes casos:

1. Colisão com outro robô do time adversário.
2. É permitido empurrar o goleiro adversário na área do gol se a bola está entre o robô que empurra e o goleiro. Entretanto, empurrar o goleiro com a bola para o gol não é permitido.
3. Atacar com mais de um robô na área do gol do time oposto será penalizado por um chute a gol a ser tomado pelo time do goleiro.
4. Defender com mais de um robô na área será marcado um pênalti.
5. É considerado como manipulação da bola quando um robô que não seja o goleiro captura a bola. Outro caso é quando um robô prende firmemente a bola não permitindo assim que outros robôs a manipulem.
6. O goleiro deverá chutar a bola de sua área em um tempo de 10s. Caso isso não seja respeitado, será marcado um pênalti para o time adversário.
7. Chutar a bola para o gol e bloquear intencionalmente o goleiro implicará em pênalti.
8. Somente o árbitro e um dos membros humanos da equipe poderão tocar os robôs. Caso não haja respeito a essa regra, será marcado um pênalti para o time adversário.

Interrupções

O jogo será interrompido e os robôs serão recolocados por operadores humanos somente quando:

1. Um robô tem que ser mudado.
2. Um robô caiu fazendo uma defesa.
3. Um gol é marcado ou uma falta ocorre.
4. O árbitro chama chute a gol ou bola livre.

Chute Livre

Quando um robô defensor intencionalmente empurra um robô oponente, um chute livre é dado ao time adversário. A bola será colocada na posição relativa ao chute livre (FK) no campo (ver figura A.1). O robô que dará o chute será colocado atrás da bola. A posição do time atacante é livre em seu próprio lado. Os robôs na defesa serão colocados na área do gol ou no arco que o forma.

Pênalti

Um pênalti será marcado nas seguintes condições:

1. Defender com mais de um robô na área do gol;
2. Falha por parte do goleiro em chutar a bola em um tempo inferior a 10s;
3. Quando alguém dos membros humanos tocarem os robôs sem a permissão do juiz, enquanto o jogo está em andamento.

Chute a Gol

Um chute a gol será marcado nas seguintes condições:

1. Quando um robô atacando empurra o goleiro em sua área do gol;
2. Atacando com mais de um robô na área do gol do time adversário;
3. Quando há uma interrupção intencional do goleiro por parte do time adversário;
4. Quando o goleiro captura a bola em sua própria área do gol;
5. Quando uma interrupção ocorre na área do gol por mais de 10s.

Bola Livre

O árbitro chamará bola livre quando uma interrupção ocorre por 10s fora da área do gol.

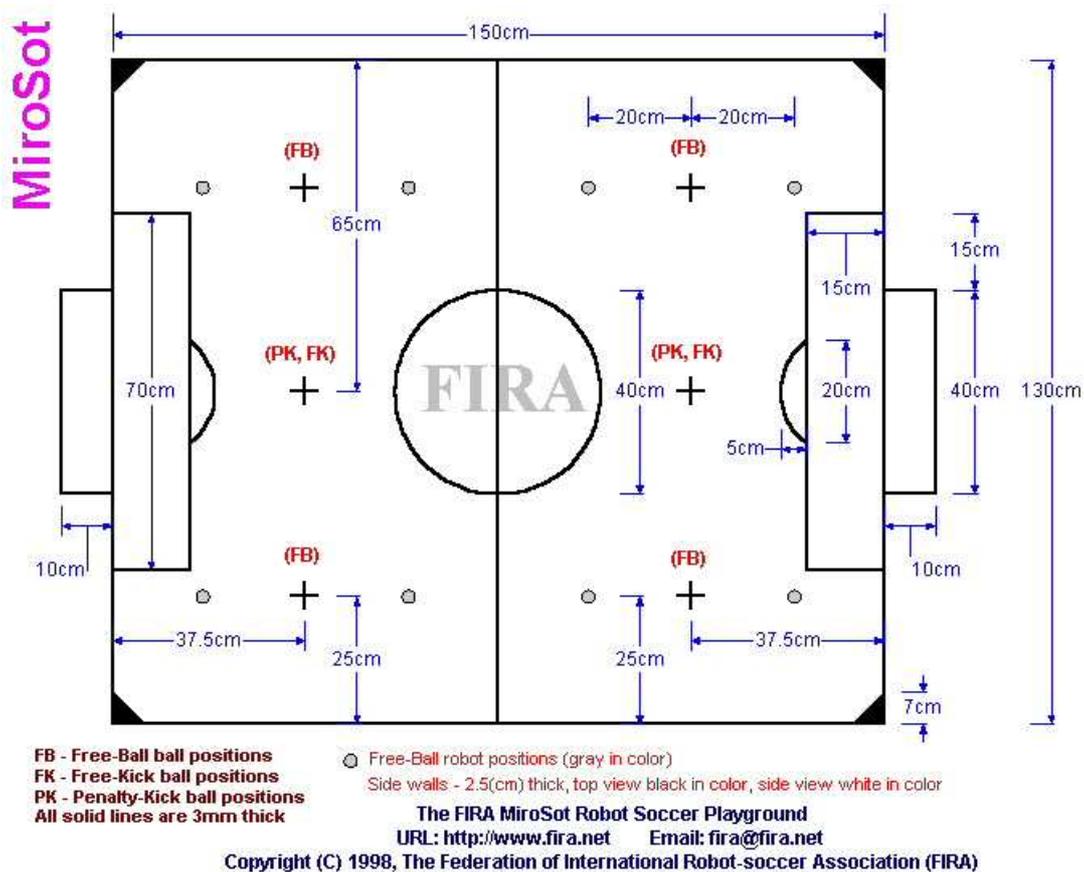


Figura A.1: Medidas padrões do campo para a categoria MiraSot-FIRA

Apêndice B

Equações Paramétricas de Terceiro Grau

B.1 Cálculo dos Parâmetros

As variáveis de configuração (x, y, θ) do minirobô são parametrizadas em λ , como mostra as equações B.1, sendo que $0 \leq \lambda \leq 1$. A equação $d(\lambda) = \tan(\theta(\lambda))$ foi introduzida para facilitar a notação.

$$\begin{aligned}x(\lambda) &= a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3 \\y(\lambda) &= b_0 + b_1\lambda + b_2\lambda^2 + b_3\lambda^3 \\d(\lambda) = \tan(\theta(\lambda)) &= \frac{b_1 + 2b_2\lambda + 3b_3\lambda^2}{a_1 + 2a_2\lambda + 3a_3\lambda^2}\end{aligned}\tag{B.1}$$

As condições de contorno para estas equações são:

$$\begin{aligned}(x(0), y(0), \theta(0)) &= (x_i, y_i, \theta_i) \quad \text{para } \lambda = 0 \\(x(1), y(1), \theta(1)) &= (x_f, y_f, \theta_f) \quad \text{para } \lambda = 1\end{aligned}$$

Assim, para encontrar os valores dos coeficientes $(a_0, \dots, a_3, b_0, \dots, b_3)$, calcula-se uma solução para o seguinte sistema:

$$\begin{cases}x(0) = a_0 \\y(0) = b_0 \\d(0) = \tan \theta(0) = \frac{b_1}{a_1} \\x(1) = a_0 + a_1 + a_2 + a_3 \\y(1) = b_0 + b_1 + b_2 + b_3 \\d(1) = \tan \theta(1) = \frac{b_1 + 2b_2 + 3b_3}{a_1 + 2a_2 + 3a_3}\end{cases}$$

Realizando algumas substituições, o sistema fica igual a:

$$\begin{cases}d_i a_1 - b_1 = 0 \\a_1 + a_2 + a_3 = \Delta x \\b_1 + b_2 + b_3 = \Delta y \\d_f a_1 + 2d_f a_2 + 3d_f a_3 - b_1 - 2b_2 - 3b_3 = 0\end{cases}$$

onde $\Delta x = x_f - x_i$ e $\Delta y = y_f - y_i$. Para efeitos de cálculo estes termos serão considerados sempre positivos, o que não implica em perda de generalidade.

Escrevendo o sistema na forma de equação matricial, tem-se:

$$\begin{bmatrix} 0 & 0 & -1 & 0 & 0 & d_i \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ -3 & -2 & -1 & 3d_f & 2d_f & d_f \end{bmatrix} \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ a_3 \\ a_2 \\ a_1 \end{bmatrix} = \begin{bmatrix} 0 \\ \Delta x \\ \Delta y \\ 0 \end{bmatrix}$$

que, aplicando o método da pivotação parcial, chega-se a:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3d_f & 2d_f & d_f \\ 0 & 0 & 1 & 0 & 0 & -d_i \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ a_3 \\ a_2 \\ a_1 \end{bmatrix} = \begin{bmatrix} \Delta y \\ 3\Delta y \\ 0 \\ \Delta x \end{bmatrix}$$

Uma vez que este sistema possui mais incógnitas que equações, deve-se escolher duas variáveis para terem seus valores arbitrados. Assim, escolhem-se as variáveis a_1 e a_2 como tais porque estes coeficientes fazem com que os outros sejam determinados sem que haja divisões, o que poderia causar singularidades.

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 2 & 3d_f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_3 \\ b_2 \\ b_1 \\ a_3 \end{bmatrix} = \begin{bmatrix} \Delta y \\ 3\Delta y \\ 0 \\ \Delta x \end{bmatrix} - \begin{bmatrix} 0 \\ 2d_f \\ 0 \\ 1 \end{bmatrix} a_2 - \begin{bmatrix} 0 \\ d_f \\ -d_i \\ 1 \end{bmatrix} a_1$$

Encontrando uma solução para este sistema tem-se:

$$\begin{cases} b_3 = 3d_f \Delta x - 2\Delta y - d_f a_2 - (2d_f - d_i) a_1 \\ b_2 = 3\Delta y - 3d_f \Delta x + d_f a_2 - 2(d_i - d_f) a_1 \\ b_1 = d_i a_1 \\ a_3 = \Delta x - a_2 - a_1 \end{cases} \quad (\text{B.2})$$

que é a forma geral para calcular os parâmetros dos polinômios de (x, y, θ) dadas as condições de contorno citadas.

Quaisquer valores atribuídos a estas variáveis fazem com que caminhos que respeitam a restrição não-holonômica sejam determinados. Porém, esta solução é válida apenas para o caso onde os ângulos θ_i e θ_f são diferentes de $\pm\pi/2$. Quando estes ângulos são iguais a $\pm\pi/2$, ocorrem singularidades em $d(\lambda)$ (equação B.1), uma vez que $\tan(\theta_i) \rightarrow \infty$ e $\tan(\theta_f) \rightarrow \infty$. Isto resulta em casos especiais que serão abordados nas próximas seções.

B.2 Casos Especiais

B.2.1 Primeiro Caso

Este primeiro caso especial considera que $\theta_i = \pm\pi/2$ e $\theta_f = \pm\pi/2$. Nestas situações, as tangentes dos ângulos tendem para infinito, ou seja, têm-se que $d_i \rightarrow \pm\infty$ e $d_f \rightarrow \pm\infty$.

Considerando as mesmas condições de contorno adotadas anteriormente, tem-se o sistema:

$$\begin{aligned}
 x(0) &= a_0 \\
 y(0) &= b_0 \\
 d(0) &\rightarrow \infty \\
 x(1) &= a_0 + a_1 + a_2 + a_3 \\
 y(1) &= b_0 + b_1 + b_2 + b_3 \\
 d(1) &\rightarrow \infty
 \end{aligned}$$

Sendo $d(0) = d_i \rightarrow \infty$, tem-se que $a_1 = 0$, de acordo com $d(\lambda)$ apresentado nas equações B.1. Da mesma forma, tem-se que $d(1) = d_f \rightarrow \infty$, fazendo com que $a_1 + 2a_2 + 3a_3 = 0$. Considerando b_1 e b_2 como parâmetros livres, pois não implicam em divisões na determinação dos outros coeficientes, e realizando algumas operações aritméticas, encontra-se a seguinte solução:

$$\begin{cases}
 a_0 = x_i \\
 a_1 = 0 \\
 a_2 = 3\Delta x \\
 a_3 = -2\Delta x \\
 b_0 = y_i \\
 b_3 = \Delta y - b_1 - b_2
 \end{cases} \quad (\text{B.3})$$

B.2.2 Segundo Caso

Aqui, considera-se que apenas $\theta_i = \pm\frac{\pi}{2}$ enquanto o ângulo final pode ter qualquer valor, exceto $\pm\frac{\pi}{2}$. De maneira similar ao primeiro caso, $d_i \rightarrow \pm\infty$ o que implica em $a_1 = 0$, de acordo com $d(\lambda)$ (equação B.1). O sistema resultante é então:

$$\begin{aligned}
 x(0) &= a_0 \\
 y(0) &= b_0 \\
 d(0) &\rightarrow \infty \\
 x(1) &= a_0 + a_1 + a_2 + a_3 \\
 y(1) &= b_0 + b_1 + b_2 + b_3 \\
 d(1) &= \frac{b_1 + 2b_2 + 3b_3}{a_1 + 2a_2 + 3a_3}
 \end{aligned}$$

Considerando a_3 e b_3 como variáveis a serem arbitradas, a solução para este novo sistema é dada pelas equações B.4.

$$\begin{cases}
 a_0 = x_i \\
 a_1 = 0 \\
 a_2 = \Delta x - a_3 \\
 b_0 = y_i \\
 b_1 = 2\Delta y - 2d_f\Delta x - d_f a_3 + b_3 \\
 b_2 = 2d_f\Delta x - \Delta y + d_f a_3 - 2b_3
 \end{cases} \quad (\text{B.4})$$

B.2.3 Terceiro Caso

Por fim, para esta situação tem-se apenas que $\theta_f = \pm \frac{\pi}{2}$. Assim, $d_f \rightarrow \pm \infty$ e de acordo com $d(\lambda)$ das equações B.1, o sistema obtido é:

$$\begin{aligned} x(0) &= a_0 \\ y(0) &= b_0 \\ d(0) &= \frac{b_1}{a_1} \\ x(1) &= a_0 + a_1 + a_2 + a_3 \\ y(1) &= b_0 + b_1 + b_2 + b_3 \\ d(1) &= \rightarrow \infty \end{aligned}$$

Adotando os coeficientes a_1 e b_2 como variáveis livres, tem-se a solução dada pelas equações B.5.

$$\begin{cases} a_0 = x_i \\ a_2 = 3\Delta x - 2a_1 \\ a_3 = -2\Delta x + a_1 \\ b_0 = y_i \\ b_1 = d_i a_1 \\ b_3 = \Delta y - d_i a_1 - b_2 \end{cases} \quad (\text{B.5})$$

Da mesma forma que para o caso geral, quaisquer valores atribuídos às variáveis livres escolhidas implicam em caminhos que respeitam a restrição não-holonômica do minirobô. Entretanto, deve-se definir um critério de escolha para o valor destes coeficientes livres de modo que os caminhos sejam considerados apropriados de acordo com o senso comum.

B.3 Determinação dos Coeficientes Livres

Um critério de determinação dos valores dos coeficientes livres deve ser adotado de forma que valores adequados sejam encontrados, e com isso, os caminhos sejam aprimorados. O critério utilizado é que os polinômios $x(\lambda)$ e $y(\lambda)$, dados pelas equações B.1 não tenham valores de máximo ou mínimo entre o intervalo $0 < \lambda < 1$. Como consequência, o caminho encontrado, sempre que possível, fica confinado em um retângulo delimitado pelas posições inicial e final, como exemplificado na figura B.1.

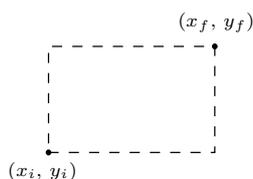


Figura B.1: Retângulo delimitado pelas posições inicial e final

Para isto, as derivadas em relação a λ destes polinômios não podem ter raízes reais neste intervalo, o que implica em quatro condições:

1. as raízes são complexas; ou

2. as raízes são reais e menores ou iguais a 0; ou
3. as raízes são reais e maiores ou iguais a 1; ou
4. uma raiz real é maior ou igual a 1 e outra, menor ou igual a 0.

Sendo as derivadas dos polinômios $x(\lambda)$ e $y(\lambda)$ dadas por:

$$\begin{aligned} \frac{d}{d\lambda}x(\lambda) &= a_1 + 2a_2\lambda + 3a_3\lambda^2 \\ \frac{d}{d\lambda}y(\lambda) &= b_1 + 2b_2\lambda + 3b_3\lambda^2 \end{aligned} \tag{B.6}$$

Considerando-se um polinômio genérico também de segundo grau, para facilitar as análises das condições impostas tem-se:

$$\lambda^2 + b\lambda + c = 0 \tag{B.7}$$

Para satisfazer a primeira condição é preciso que o discriminante da equação B.7 seja menor que zero. Assim:

$$b^2 - 4c < 0 \rightarrow c > \frac{b^2}{4}$$

que em representação gráfica, é a área assinalada na figura B.2.

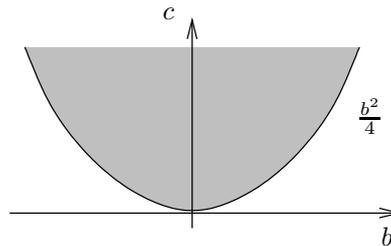


Figura B.2: Região para a qual os valores de b e c implicam em raízes complexas para a equação B.7

Para que a segunda condição imposta seja satisfeita, utiliza-se a análise do critério de estabilidade de Routh-Hurwitz [Oga97].

$$\begin{array}{r} \lambda^2 \quad 1 \quad c \\ \lambda^1 \quad b \\ \lambda^0 \quad c \end{array}$$

Para que a equação B.7 tenha raízes menores ou iguais a 0, é preciso que $b \geq 0$ e $c \geq 0$. Graficamente, isto é apresentado na figura B.3.

Para a terceira condição utiliza-se também o critério de Routh, mas com algumas modificações na variável. Sendo $\gamma = 1 - \lambda$, quando $\lambda \geq 1$ implicará em $\gamma \leq 0$. Dessa forma, substituindo $\lambda = 1 - \gamma$ na equação B.7 tem-se:

$$\gamma^2 - (2 + b)\gamma + (1 + b + c) = 0$$

Aplicando o algoritmo de Routh, resulta:

$$\begin{array}{r} \gamma^2 \quad 1 \quad (1 + b + c) \\ \gamma^1 \quad -(2 + b) \\ \gamma^0 \quad (1 + b + c) \end{array}$$

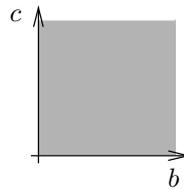


Figura B.3: Região na qual os valores dos coeficientes b e c implicam em raízes reais menores ou iguais a 0 para a equação B.7

Tem-se que $-(2 + b) \geq 0$ e $(1 + b + c) \geq 0$ para que as raízes da equação B.7 sejam reais e maiores ou iguais a 1. Dessa forma:

$$\begin{aligned} -(2 + b) \geq 0 &\rightarrow b \leq -2 \\ (1 + b + c) \geq 0 &\rightarrow c \leq -b - 1 \end{aligned}$$

A região assinalada para esta condição é dada pela figura B.4.

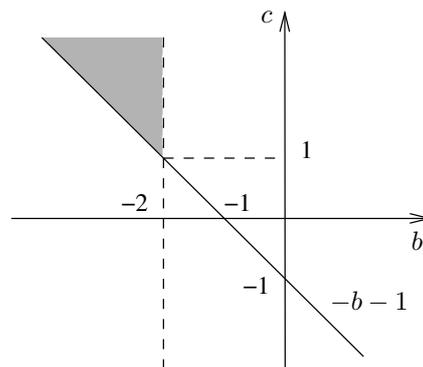


Figura B.4: Região de valores dos coeficientes b e c da equação B.7 que implicam em raízes reais maiores ou iguais a 1

Para a última condição, realiza-se uma interseção entre os dois procedimentos anteriores, ou seja, aplica-se o critério de Routh de modo que a equação B.7 tenha uma raiz real menor ou igual a 0 e outra, maior ou igual a 1.

$$\begin{array}{ccc} \lambda^2 & 1 & c \\ \lambda^1 & b & \\ \lambda^0 & c & \end{array} \qquad \begin{array}{ccc} \gamma^2 & 1 & (1 + b + c) \\ \gamma^1 & -(2 + b) & \\ \gamma^0 & (1 + b + c) & \end{array} \qquad (B.8)$$

Assim, sendo $c \leq 0$ e $c \leq -b - 1$, haverá apenas uma variação de sinal nos coeficientes de Routh, satisfazendo a condição. A figura B.5 mostra a região encontrada.

Por fim, realizando a união destas quatro áreas encontradas, tem-se uma região na qual os valores dos coeficientes permitem que os critérios desejados sejam atendidos para um polinômio de segundo grau. Esta região final, visualizada na figura B.6, é delimitada pelas condições B.9.

$$[(c \geq 0) \cap (c \geq -b/2) \cap (c \geq -b - 1)] \cup [(c \leq 0) \cap (c \leq -b - 1)] \qquad (B.9)$$

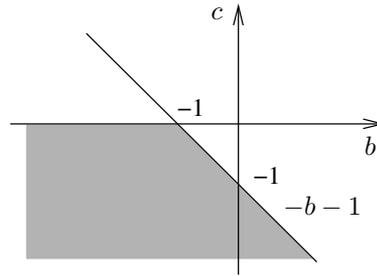


Figura B.5: Região para que a equação B.7 tenha uma raiz real menor ou igual a 0, e outra maior ou igual a 1

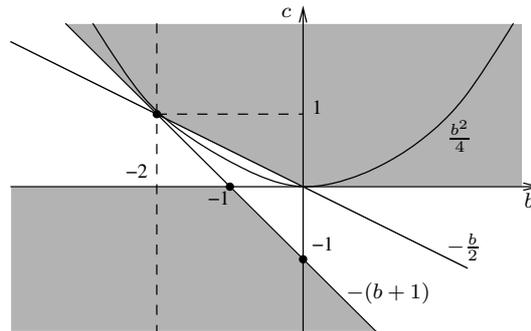


Figura B.6: Região para que a equação B.7 não tenha valores máximos ou mínimos entre $0 < \lambda < 1$

A reta $c = -b/2$ foi utilizada de forma a permitir uma maior simplicidade nos cálculos, uma vez que ela substitui o termo $c = \frac{b^2}{4}$ sem perder a generalidade. Uma vez encontrada a região delimitada pelas condições B.9, deve-se analisar separadamente para os polinômios nas direções x e y , tanto para o caso geral quanto para os casos especiais.

B.3.1 Caso Geral

Análise na Direção x

Para análise das condições impostas para satisfazer o critério de aprimoramento estabelecido na direção x , considera-se a equação B.6. Igualando-a $dx/d\lambda$ a 0, tem-se:

$$\lambda^2 + \frac{2a_2}{3a_3} + \frac{a_1}{3a_3} = 0$$

Utilizando a região final delimitada pelas condições B.9, tem-se para $c \geq 0$:

$$\frac{a_1}{3a_3} \geq 0$$

que implica na região delimitada pelas condições B.10. Considerando o termo Δx positivo, a figura B.7 apresenta graficamente a região obtida.

$$[(a_1 \geq 0) \cap (a_2 < \Delta x - a_1)] \cup [(a_1 \leq 0) \cap (a_2 > \Delta x - a_1)] \quad (\text{B.10})$$

De forma complementar, para $c \leq 0$, tem-se a região delimitada pelas condições B.11 e visualizada na figura B.8.

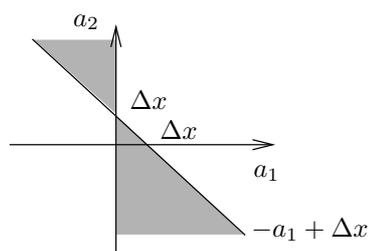


Figura B.7: Região delimitada pelas condições B.10

$$[(a_1 \geq 0) \cap (a_2 > \Delta x - a_1)] \cup [(a_1 \leq 0) \cap (a_2 < \Delta x - a_1)] \quad (\text{B.11})$$

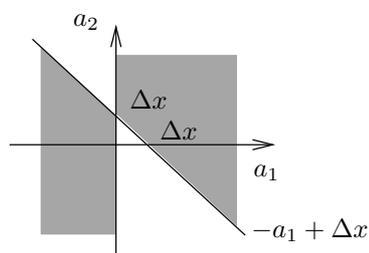


Figura B.8: Região delimitada pelas condições B.11

Para $c \geq -b/2$, a tem-se a inequação:

$$\frac{a_1}{3a_3} \geq -\frac{2a_2}{6a_3} \rightarrow \frac{2a_1 + 2a_2}{6a_3} \geq 0$$

que fazendo as devidas substituições, chega-se as condições B.12 que delimitam a região mostrada na figura B.9.

$$[(a_2 \geq -a_1) \cap (a_2 < \Delta x - a_1)] \cup [(a_2 \leq -a_1) \cap (a_2 > \Delta x - a_1)] \quad (\text{B.12})$$

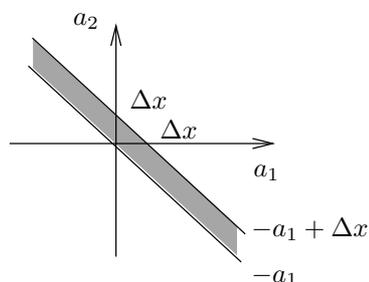


Figura B.9: Região delimitada pelas condições B.12

Para $c \geq -b - 1$, tem-se:

$$\frac{a_1}{3a_3} \geq -\frac{2a_2}{3a_3} - 1 \rightarrow \frac{a_1 + 2a_2 + 3a_3}{3a_3} \geq 0$$

Substituindo os termos e efetuando os cálculos necessários, chega-se a uma região delimitada pelas condições B.13 e apresentada na figura B.10.

$$[(a_2 \leq 3\Delta x - 2a_1) \cap (a_2 < \Delta x - a_1)] \cup [(a_2 \geq 3\Delta x - 2a_1) \cap (a_2 > \Delta x - a_1)] \quad (\text{B.13})$$

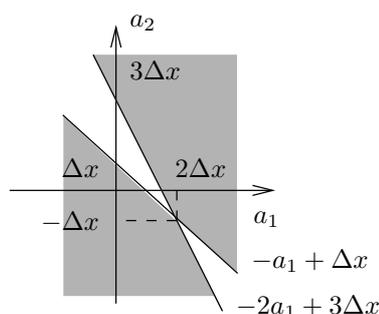


Figura B.10: Região delimitada pelas condições B.13

Por fim, também de forma complementar, para $c \leq -b - 1$ tem-se a região delimitada pelas condições B.14 e visualizada na figura B.11.

$$[(a_2 \leq 3\Delta x - 2a_1) \cap (a_2 > \Delta x - a_1)] \cup [(a_2 \geq 3\Delta x - 2a_1) \cap (a_2 < \Delta x - a_1)] \quad (\text{B.14})$$

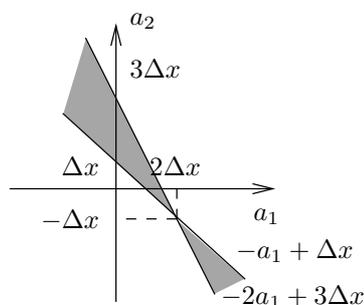


Figura B.11: Região delimitada pelas condições B.14

Para encontrar a região de valores admissíveis para a_1 e a_2 para que não haja pontos de máximo ou mínimo na direção x , entre o intervalo fechado $[0, 1]$ de λ , deve-se proceder com as uniões e interseções das regiões delimitada pelas condições B.10 até B.14, segundo as condições B.9. Esta região final, visualizada na figura B.12, pode ser descrita através das condições B.15.

$$\begin{cases} a_1 \geq 0 \\ a_2 \geq -a_1 \\ a_2 \leq 3\Delta x - 2a_1 \end{cases} \quad (\text{B.15})$$

Portanto, quaisquer valores para a_1 e a_2 nesta região final garantem que o polinômio $x(\lambda)$ das equações B.1 não tenha valores máximos ou mínimos entre $0 < \lambda < 1$.

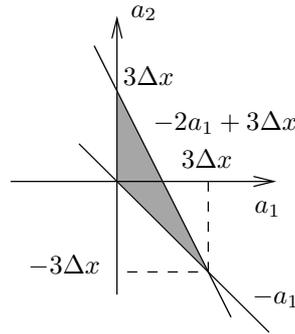


Figura B.12: Região final dos coeficientes livres do caso geral para satisfazer o critério de aprimoramento na direção x

Análise na Direção y

Sendo os polinômios de $y(\lambda)$ e $x(\lambda)$ semelhantes na forma, a região final para o critério na direção y pode ser descrita através das condições:

$$\begin{cases} b_1 \geq 0 \\ b_2 \geq -b_1 \\ b_2 \leq 3\Delta y - 2b_1 \end{cases}$$

Considerando Δy positivo, a região pode ser visualizada na figura B.13.

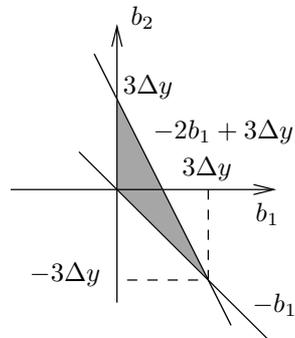


Figura B.13: Região final admissível para $y(\lambda)$

Substituindo os coeficientes b_1 e b_2 de acordo com a equação B.2, têm-se:

$$\begin{cases} d_i a_1 \geq 0 \\ d_f a_2 \geq (d_i - 2d_f) a_1 + 3(df \Delta x - \Delta y) \\ d_f a_2 \leq 3d_f \Delta x - 2d_f a_1 \end{cases} \quad (\text{B.16})$$

Percebe-se que as condições B.16, além dos coeficientes livres, dependem também dos termos Δx , Δy , d_i e d_f . Porém, como sempre é suposto que $\Delta x > 0$ e $\Delta y > 0$ sem que isso afete a análise do critério, oito condições distintas para atender a critério de aprimoramento na direção y são definidas:

1. $d_i > 0$ e $d_f > 0$;

2. $d_i > 0$ e $d_f < 0$;
3. $d_i < 0$ e $d_f > 0$;
4. $d_i < 0$ e $d_f < 0$;
5. $d_i = 0$ e $d_f > 0$;
6. $d_i = 0$ e $d_f < 0$;
7. $d_i > 0$ e $d_f = 0$;
8. $d_i < 0$ e $d_f = 0$;
9. $d_i = 0$ e $d_f = 0$.

Para o caso de $d_i > 0$ e $d_f > 0$ tem-se que a região, visualizada na figura B.14, é delimitada pelas condições B.17.

$$\begin{cases} a_1 \geq 0 \\ a_2 \geq \frac{d_i - 2d_f}{d_f} a_1 + 3 \frac{d_f \Delta x - \Delta y}{d_f} \\ a_2 \leq -2a_1 + 3\Delta x \end{cases} \quad (\text{B.17})$$

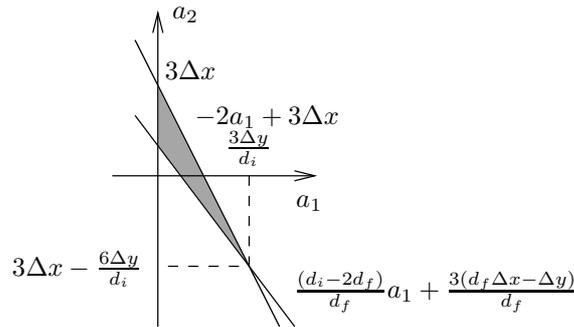


Figura B.14: Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f > 0$

Para o caso de $d_i > 0$ e $d_f < 0$ tem-se a região delimitada pelas condições B.18 e visualizada na figura B.15.

$$\begin{cases} a_1 \geq 0 \\ a_2 \leq \frac{d_i - 2d_f}{d_f} a_1 + 3 \frac{d_f \Delta x - \Delta y}{d_f} \\ a_2 \geq -2a_1 + 3\Delta x \end{cases} \quad (\text{B.18})$$

Para o caso de $d_i < 0$ e $d_f > 0$ tem-se a região delimitada pelas condições B.19 e visualizada na figura B.16.

$$\begin{cases} a_1 \leq 0 \\ a_2 \geq \frac{d_i - 2d_f}{d_f} a_1 + 3 \frac{d_f \Delta x - \Delta y}{d_f} \\ a_2 \leq -2a_1 + 3\Delta x \end{cases} \quad (\text{B.19})$$

Para o caso de $d_i < 0$ e $d_f < 0$ tem-se as condições B.20, visualizadas na figura B.17.

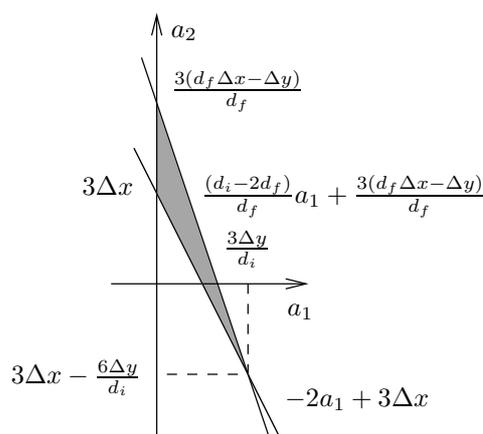


Figura B.15: Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f < 0$

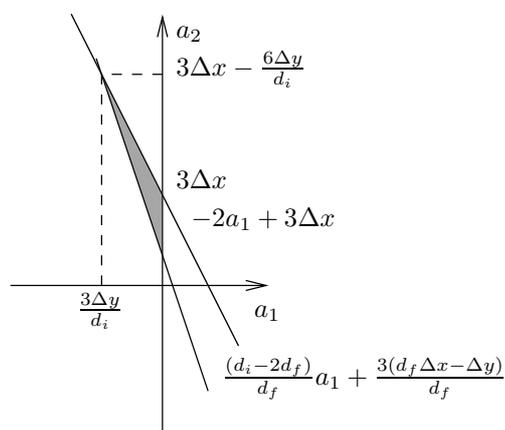


Figura B.16: Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f > 0$

$$\begin{cases} a_1 \leq 0 \\ a_2 \leq \frac{d_i - 2d_f}{d_f} a_1 + 3 \frac{d_f \Delta x - \Delta y}{d_f} \\ a_2 \geq -2a_1 + 3\Delta x \end{cases} \quad (\text{B.20})$$

Para o caso $d_i > 0$ e $d_f = 0$ tem-se a região delimitada pelas condições B.21, visualizada na figura B.18.

$$0 \leq a_1 \leq \frac{3\Delta y}{d_i} \quad (\text{B.21})$$

Para o caso de $d_i < 0$ e $d_f = 0$ tem-se a região delimitada pelas condições B.22 e visualizada na figura B.19.

$$\frac{3\Delta y}{d_i} \leq a_1 \leq 0 \quad (\text{B.22})$$

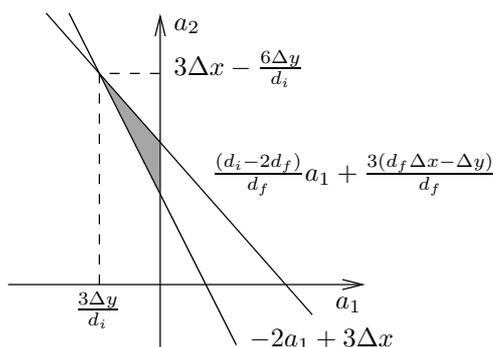


Figura B.17: Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f < 0$

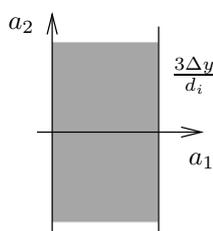


Figura B.18: Região admissível para não haver máximos e mínimos na direção y quando $d_i > 0$ e $d_f = 0$

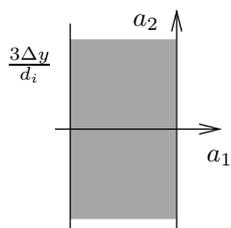


Figura B.19: Região admissível para não haver máximos e mínimos na direção y quando $d_i < 0$ e $d_f = 0$

Para o caso de $d_i = 0$ e $d_f > 0$ tem-se a região delimitada pelas condições B.23 e visualizada na figura B.20.

$$\begin{cases} a_2 \leq -2a_1 + 3\Delta x \\ a_2 \geq -2a_1 + \frac{3(d_f\Delta x - \Delta y)}{d_f} \end{cases} \quad (\text{B.23})$$

Para o caso de $d_i = 0$ e $d_f < 0$ tem-se a região delimitada pelas condições B.24 e visualizada na figura B.21.

$$\begin{cases} a_2 \geq -2a_1 + 3\Delta x \\ a_2 \leq -2a_1 + \frac{3(d_f\Delta x - \Delta y)}{d_f} \end{cases} \quad (\text{B.24})$$

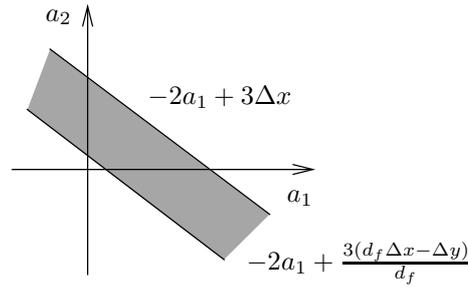


Figura B.20: Região admissível para não haver máximos e mínimos na direção y quando $d_i = 0$ e $d_f > 0$

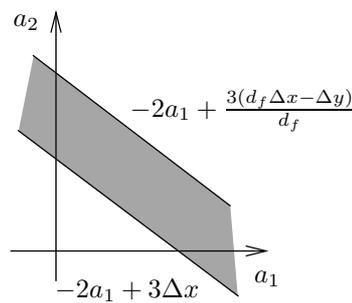


Figura B.21: Região admissível para não haver máximos e mínimos na direção y quando $d_i = 0$ e $d_f < 0$

E, finalmente, para o caso onde $d_i = 0$ e $d_f = 0$, a região de valores admissíveis para a_1 e a_2 para não haver máximos e mínimos na direção y não é definida, implicando que quaisquer valores atribuídos para estas variáveis irão satisfazer este caso em particular.

Para que valores únicos sejam atribuídos para a_1 e a_2 de forma que o caminho, simultaneamente nas direções x e y , não apresente valores máximo ou mínimo é necessário que haja uma interseção entre a região delimitada para a direção x e cada região encontrada para a direção y , dependendo dos valores de θ_i e θ_f , e que um ponto desta interseção seja escolhido.

B.3.2 Casos Especiais

Análise para $\theta_i = \pm\pi/2$ e $\theta_f = \pm\pi/2$

Para esta situação, tem-se b_1 e b_2 como parâmetros livres. No entanto, somente o polinômio $y(\lambda)$ depende destes valores. Para conseguir encontrar os valores dos parâmetros livres procede-se de maneira similar ao realizado com $y(\lambda)$ do caso geral, porém utilizando as equações B.3. Assim, a região de valores dos coeficientes livres b_1 e b_2 que satisfazem ao critério na direção y é delimitada pelas inequações:

$$\begin{cases} b_1 \geq 0 \\ b_2 \geq -b_1 \\ b_2 \leq 3\Delta y - 2b_1 \end{cases}$$

que, considerando Δy positivo, pode ser visualizada na figura B.22.

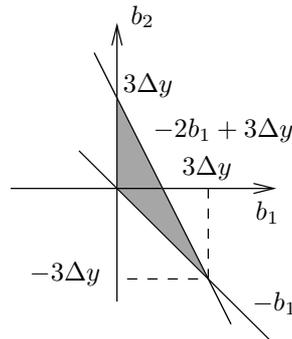


Figura B.22: Região final admissível para o caso onde θ_i e θ_f são iguais a $\pm\pi/2$

Análise para $\theta_i = \pm\pi/2$ e $\theta_f \neq \pm\pi/2$

Para o caso onde somente $\theta_i = \pm\pi/2$, tem-se a_3 e b_3 como coeficientes de valores livres, como apresentado nas equações B.4. Aplicando o critério de aprimoramento na direção x para estas equações, tem-se que a região final é delimitada por:

$$-2\Delta x \leq a_3 \leq \Delta x$$

que, considerando Δx positivo, pode ser visualizada na figura B.23.

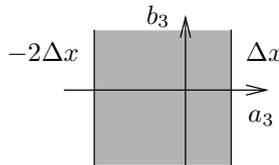


Figura B.23: Região final admissível na direção x . Caso onde somente $\theta_i = \pm\pi/2$

De maneira similar, aplicando o critério de aprimoramento na direção y tem-se uma região delimitada por:

$$\begin{cases} d_f a_3 \geq -2d_f \Delta x \\ b_3 \geq d_f a_3 - 2(\Delta y - d_f \Delta x) \\ b_3 \leq \Delta y \end{cases}$$

que considerando d_f , Δx e Δy positivos pode ser visualizada na figura B.24.

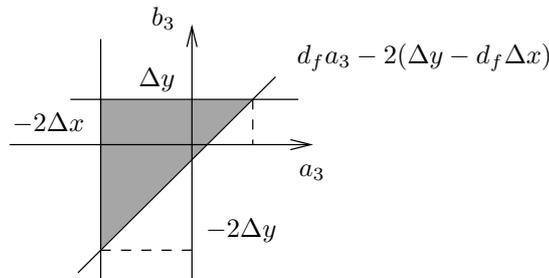


Figura B.24: Região final admissível na direção y . Caso onde somente $\theta_i = \pm\pi/2$

Dessa forma, para que se possa, nestas condições, encontrar um caminho que satisfaça o critério de aprimoramento imposto simultaneamente nas direções x e y é preciso que a_3 e b_3 sejam escolhidos a partir da interseção, quando isto for possível, das regiões exemplificadas nas figuras B.23 e B.24.

Análise para $\theta_i \neq \pm\pi/2$ e $\theta_f = \pm\pi/2$

Por fim, para o caso onde somente $\theta_f = \pm\pi/2$ tem-se os coeficientes livres a_1 e b_2 , de acordo com as equações B.5. Para a direção x , o critério de aprimoramento do caminho para o robô implica na seguinte região:

$$0 \leq a_1 \leq 3\Delta x$$

que pode ser visualizada na figura B.25, caso $\Delta x > 0$.

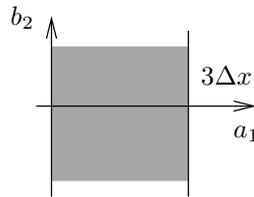


Figura B.25: Região final admissível na direção x . Caso onde somente $\theta_f = \pm\pi/2$

Da mesma maneira, aplicando o critério de aprimoramento do caminho na direção y tem-se a região final delimitada por:

$$\begin{cases} d_i a_1 \geq 0 \\ b_2 \geq -d_i a_1 \\ b_2 \leq -2d_i a_1 + 3\Delta y \end{cases}$$

Considerando d_i e Δy maiores que 0, esta região pode ser visualizada na figura B.26.

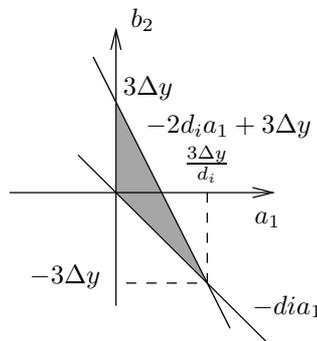


Figura B.26: Região final admissível na direção y . Caso onde somente $\theta_i = \pm\pi/2$

Caso seja possível, realiza-se a interseção entre as regiões exemplificadas nas figuras B.25 e B.26 e obtêm-se valores para a_1 e b_2 de forma a aprimorar o caminho tanto na direção x quanto y .

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)