

Daniel José Marcelino

Navegação de Robôs Móveis em Formação

Setembro 2016



UNIVERSIDADE DE COIMBRA



FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Navegação de Robôs Móveis em Formação

Daniel José Marcelino

Coimbra, Setembro 2016



Navegação de Robôs Móveis em Formação

Orientador:

Prof. Doutor Rui Paulo Pinto da Rocha

Júri:

Prof. Doutor Lino José Forte Marques

Prof. Doutor Sérgio Paulo Carvalho Monteiro

Prof. Doutor Rui Paulo Pinto da Rocha

Dissertação submetida como requisito parcial para obtenção do grau de Mestre em
Engenharia Eletrotécnica e de Computadores.

Coimbra, Setembro 2016

Agradecimentos

O presente trabalho não seria bem conseguido sem o suporte de várias pessoas.

Ao Professor Rui Rocha, agradeço a sua partilha de sabedoria e experiência para comigo, assim como toda a disponibilidade que teve para me orientar.

À Daniela Nobre, Diogo Amorim, João Rosa e Rui Carvalho, um obrigado pela partilha de conhecimento durante o nosso percurso pelo AP4ISR.

Ao Gonçalo Martins e Paulo Ferreira, obrigado pela disponibilidade que demonstraram quando precisei.

Ao André Conceição, Diogo Amorim, Ivo Ferreira, Rui Carvalho e Urbano Nunes, obrigado por ter tido possibilidade de partilhar conhecimento ROS e derivados convosco.

À Catarina Francisco, obrigado por me teres dado suporte sempre que foi necessário, mas principalmente por me conseguires manter sempre calmo e confiante.

À minha irmã, obrigado por me ajudares a encontrar as palavras certas.

Aos meus pais, obrigado por tudo.

Resumo

Ao longo dos últimos anos o estudo de metodologia para o controlo de formação tem vindo a ser alvo de grande atenção por parte de muitas equipas de investigação. Este interesse advém da vasta aplicabilidade que a navegação de um grupo de robôs em formação tem.

Após se realizar um estudo sobre as diferentes metodologias existentes na literatura, optou-se pela escolha de uma estratégia baseada em comportamentos, utilizando dinâmicas de atração. Este tipo de dinâmicas, dividem-se em dinâmicas de atração e de repulsão, onde uma dinâmica de atração pode ser dada pela posição objetivo a ser tomada pelo robô e a dinâmica de repulsão pode ser originada pelos obstáculos que rodeiam o robô.

A dinâmica de repulsão é determinada com recurso a leituras provenientes de um Medidor de Distância a Laser, onde cada leitura gera uma contribuição para a dinâmica de repulsão. Faz-se uso de um sistema de localização relativa que tem como base o uso de uma câmara monocular instalada em cada robô, possibilitando a deteção, distinção e estimação da posição de marcadores visuais colocados nas plataformas robóticas.

Com a capacidade de estimar a posição de marcadores, torna-se possível atribuir a cada robô seguidor uma dinâmica de atração com a finalidade de seguir uma outra plataforma, mantendo uma distância e orientação fixas.

Torna-se possível desta forma a navegação de robôs móveis em formação em ambientes obstruídos, desconhecidos, e sem existência de arquitetura de rede sem fios, com o uso da aplicação distribuída implementada.

Palavras-chave: Controlo de Formação, Dinâmicas de Atração, Constrangimentos Sensoriais, Desvio de Obstáculos, Localização Relativa, Aplicação Distribuída.

Abstract

Since a few years the study on methodology for the formation control has been a huge target of attention by many investigation teams. This interest appears from the vast ways that the navigation of a robots group in formation has.

After realizing a study on different methodologies that exist in literature, was chosen a behaviour-based strategy using attractor dynamics. This kind of dynamics can be divided into dynamics of attraction and repulsion, the first one is given by the objective position being taken by the robot and the second one is lead by obstacles that surround the robot.

The dynamic of repulsion is determined by the reading of a Laser Range Finder, where each reading creates a contribute for the dynamic. It's used a relative location system based on a monocular camera installed in each robot that allows the detection, distinction and estimation of the position of visual markers placed on robotic platforms.

With the ability to estimate the markers position, becomes possible to attribute to each follower robot a dynamic of attraction with the capacity of following another platform by keeping a distance and fixed orientation.

Thus it's possible the navigation of mobile robots in formation on obstructed, unknown or without wireless network architecture environments, with the implementation of the distributed application.

Keywords: Formation Control, Attractor Dynamics, Sensor Constraints, Obstacle Avoidance, Relative Localization, Distributed Application.

“Detesto, de saída, quem é capaz de marchar em formação com prazer ao som de uma banda. Nasceu com cérebro por engano; bastava-lhe a espinhal medula.”

— Albert Einstein

Conteúdo

Agradecimentos	ii
Resumo	iii
Abstract	iv
Lista de Acrónimos	xi
Lista de Figuras	xii
Lista de Tabelas	xiv
1 Introdução	1
1.1 Navegação de Robôs Móveis em Formação	1
1.2 Motivação	2
1.3 Objetivos principais	3
1.4 Estrutura do Documento	4
2 Estratégias de Controlo de Formação	5
2.1 Abordagem Baseada em Comportamentos	5
2.2 Abordagem Líder-Seguidor	6
2.3 Abordagem baseada em Estruturas Virtuais	7
2.4 Funções de Potencial Artificial	8
2.5 Abordagem baseada na Teoria dos Grafos	9
2.6 Comparação entre Estratégias de Controlo de Formação	10
2.7 Sumário	10
3 Dinâmicas de Atração	11
3.1 Princípios Básicos das Dinâmicas de Atração	11

3.2	Aplicação das Dinâmicas de Atração para o Movimento de um Robô	12
3.2.1	Orientação do Robô	13
3.2.2	Velocidade de Deslocação	17
3.3	Sumário	19
4	Sistema Desenvolvido	21
4.1	Ambiente ROS	21
4.1.1	Nós	21
4.1.2	Mensagens	21
4.1.3	Tópicos	22
4.1.4	Serviços	22
4.2	Plataformas Robóticas	22
4.2.1	Determinação da Posição Relativa do Robô Líder	23
4.2.2	Deteção de Obstáculos	25
4.2.3	Aplicar velocidades à plataforma	26
4.3	Ambiente de Simulação	26
4.4	Notação Usada	27
4.5	Aplicação das Dinâmicas de Atração ao Controlo de Formação	28
4.5.1	Formação em Coluna	28
4.5.2	Formação Oblíqua	30
4.5.3	Integração de Comportamentos	32
4.6	Sumário	33
5	Experiências e Análise e Discussão de Resultados	35
5.1	Sistema de Cores para a Diferenciação de Robôs	35
5.2	Modelo de Apresentação de Resultados	36
5.2.1	Constituição do Erro de Distância	36
5.2.2	Constituição do Erro de Orientação	36
5.3	Experiências com 3 Robôs	37
5.3.1	Formação em Coluna	37
5.3.2	Formação Oblíqua	41
5.3.3	Restrições do Campo de Visão	44
5.4	Experiências com 12 Robôs em Ambiente de Simulação	46
5.4.1	Com Restrições de Campo de Visão	47

5.4.2	Sem Restrições de Campo de Visão	49
5.5	Sumário	50
6	Conclusão e Trabalho Futuro	51
6.1	Conclusão	51
6.2	Trabalho Futuro	52
7	Bibliografia	53

Lista de Acrónimos

FOV	Field Of View (Campo de visão)
GPS	Global Positioning System (Sistema de posicionamento global)
LRF	Laser Range Finder (Medidor de distância a laser)

Lista de Figuras

2.1	Notação dos controladores $l-\varphi$ e $l-l$	7
3.1	Pontos fixos dos sistemas dinâmicos.	12
3.2	Demonstração do ambiente que rodeia o robô.	13
3.3	Intervalo angular onde a força repulsiva atua.	14
3.4	Forma da resposta repulsiva.	15
3.5	Comparação da contribuição entre sensores de baixa e de alta resolução.	15
3.6	Forma da resposta atrativa.	16
3.7	Forma da resposta dinâmica para a orientação do robô.	17
3.8	Exemplo da função potencial da orientação do robô para evitar obstáculos.	19
4.1	Plataforma robótica utilizada.	23
4.2	Dimensões dos marcadores visuais e marcadores utilizados nas experiências em ambiente real.	24
4.3	Notação usada na dissertação.	27
4.4	Exemplo de formação em coluna.	29
4.5	Exemplo de formação em oblíquo.	30
4.6	Demonstração das contribuições existentes para a dinâmica da orientação do robô em formações do tipo oblíquo.	32
4.7	Diferentes tipos de contribuição para o controlo de formação do tipo oblíquo.	32
5.1	Informação disponível para determinação de erros.	36
5.2	Experiência para a forma geométrica do tipo coluna com a sala de testes sem obstáculos.	38
5.3	Experiência para a forma geométrica do tipo coluna com a finalidade de contornar um obstáculo.	40
5.4	Experiência para a forma geométrica do tipo coluna com ambiente obstruído.	41

5.5	Experiência para a forma geométrica do tipo oblíquo com a sala de testes sem obstáculos.	42
5.6	Experiência para a forma geométrica do tipo oblíquo com a finalidade de contornar um obstáculo.	43
5.7	Experiência para a forma geométrica do tipo oblíquo com ambiente obstruído.	45
5.8	Experiência para a forma geométrica do tipo V com ambiente obstruído. . .	46
5.9	Formação desejada para experiências com 12 robôs.	47
5.10	Percurso realizados na experiência com 12 robôs com as restrições do campo de visão.	47
5.11	Erro de distância na experiência com 12 robôs com as restrições do campo de visão.	48
5.12	Erro de orientação na experiência com 12 robôs com as restrições do campo de visão.	48
5.13	Percurso realizados na experiência com 12 robôs sem as restrições do campo de visão.	49
5.14	Erro de distância na experiência com 12 robôs sem as restrições do campo de visão.	49
5.15	Erro de orientação na experiência com 12 robôs sem as restrições do campo de visão.	50

Lista de Tabelas

2.1	Comparação entre estratégias de controlo de formação, características verificadas, não verificadas e possivelmente verificadas.	10
2.2	Comparação entre estratégias de controlo de formação, capacidade de execução das características.	10
5.1	Identificação de robôs.	35

1 Introdução

Neste capítulo é exposta a importância e as motivações para a concretização desta dissertação de mestrado. Como ponto de partida realiza-se uma breve apresentação dos princípios e características essenciais da navegação de robôs móveis em formação, assim como algumas das suas aplicações no dia a dia das pessoas. No fim do capítulo são expostos os principais objetivos a serem tomados no decorrer do trabalho.

1.1 Navegação de Robôs Móveis em Formação

Um robô móvel autónomo caracteriza-se por ter a capacidade de se movimentar e navegar autonomamente, i.e. sem intervenção humana, no ambiente para que foi desenhado, e claro está o facto de não estar fixo em nenhuma estrutura. Estes robôs podem ser desenhados para locomoção em ambientes terrestres, aéreos ou aquáticos, dependendo da aplicação para que está confinado.

A plataforma robótica é dividida em duas partes, parte mecânica e parte eletrónica. A parte mecânica que pode ser considerada como a parte física do robô, isto é, a parte que dá o aspeto e as habilidades necessárias para a sua locomoção, tais como hélices, rodas, lagartas ou pernas. Com a parte eletrónica pretende-se ter a capacidade de controlo sobre todos os atuadores aplicados ao robô na parte de mecânica, tomando em consideração o ambiente que rodeia o robô por intermédio de dispositivos sensoriais. Com placas de processamento aplicadas no robô, normalmente microcontroladores, e com a implementação de algoritmos de perceção, atuação e inteligência artificial, a plataforma robótica passa a ter um certo grau de autonomia e “inteligência”.

Neste trabalho, a “inteligência” que se pretende aplicar a todos os robôs pertencentes a uma equipa consiste na capacidade de cada robô tomar decisões, decisões estas que têm por objetivo estabelecer uma determinada disposição relativa dos robôs da equipa, i.e. uma determinada formação (*e.g.*, linha, coluna, quadrado, losango, *etc.*). Deste modo a equipa

de robôs deve conseguir realizar o movimento até ao local de destino com uma determinada formação que deve ser mantida em qualquer circunstância, exceto quando o robô estiver perante obstáculos estáticos ou dinâmicos. Neste caso, a frota de robôs deve ter a capacidade de ultrapassar estes obstáculos, sem que haja colisão com os mesmos e, depois do transitório necessário para ultrapassar a perturbação, readquirir a formação pretendida. E claro, em momento algum os robôs devem chocar entre si.

1.2 Motivação

Diversos sistemas biológicos [43], tais como cardumes de peixes ou bandos de aves, recorrem à criação de formações em prol da sua defesa aos ataques dos diversos predadores que se cruzam no seu quotidiano. Também quando uma alcateia de leões, por exemplo, pretende atacar uma presa, necessita de um grande nível de organização, recorrendo à criação de formações ao longo do ataque, para que a probabilidade da presa escapar seja escassa.

Em quaisquer exercícios militares [44], existe por base o princípio do uso de formação, para manter o maior rigor e eficácia possível no ataque ou na defesa ao inimigo. Diversos factos históricos relatam a organização das tropas nas primeiras guerras. Já nos dias de hoje, em treinos realizados por parte da força aérea com o uso de caças, é visível a constante exercitação de formações. Sugerindo assim a necessidade de implementar algoritmos de controlo de formação nos veículos utilizados pelas forças armadas [14].

No quotidiano das pessoas, estas executam por vezes atividades em que é útil o movimento de dois indivíduos em formação, *e.g.* no transporte de grandes cargas. O mesmo se passa com equipa de robôs que pretendem automatizar estas tarefas.

Com a necessidade de transporte de cargas [33], por vezes é necessário que haja um trabalho de equipa entre os robôs para que consigam transportar a carga entre locais distintos. Com uma equipa de robôs o esforço que seria necessário um robô realizar, que na maior parte dos casos é insuportável, é distribuído por todos os membros da equipa. Como é fácil de imaginar, este trabalho de equipa pode ser aplicado em ambientes industriais, como no transporte e manuseamento de objetos, ou em ambientes catastróficos como na busca de pessoas ou animais, ou até no transporte de objetos extremamente perigosos como bombas ou químicos.

Em plantações agrícolas [20], o uso de robôs favorece o desuso de tratores de grande porte em diversas atividades, diminuindo assim o impacto ambiental causado pelos gases libertados por estes veículos. Exemplos de atividades agrícolas em que pode ser explorado o

uso de equipas de robôs em formação são a plantação com uso de equipamentos terrestres e a pulverização com o uso de equipamentos aéreos.

Ao se pretender que uma frota de robôs siga em formação um objeto, pessoa ou animal, o controlo de formação pode assegurar essa tarefa. Em atividades como na formação de veículos aéreos não tripulados [29], de naves espaciais [35], ou no agrupamento de satélites [25], o uso do controlo de formação é essencial. Em diversas aplicações aquáticas é importante que um conjunto de robôs se mantenha em formação [46], como no uso de robôs aquáticos para exploração de ambientes. De igual modo, existe interesse na aplicação deste tipo de controlo em atividades relacionadas com a segurança como na deteção e captura de intrusos [13] e inclusive na exploração e reconhecimento ambiental [17].

É esperado que, com a navegação em formação de robôs móveis, todos os robôs que constituem a frota se encontrem no local destino no mesmo instante, isto é, que nenhum se tenha perdido. Desta forma é apresentada uma das grandes necessidades no quotidiano e igualmente uma das grandes vantagens do uso deste controlo de robôs.

Dependendo da obrigatoriedade de manter a formação, o controlo é diferente. No caso de ser inevitável alterar a formação, a equipa de robôs deve ter a capacidade de contornar obstáculos sem nunca alterar a formação. Contudo existem situações em que a formação, embora deva ser mantida, pode ser alterada suavemente por instantes, sem pôr em causa o objetivo final. Em certos casos, pode dividir-se a equipa em sub-equipas por forma a que seja mais fácil negociar e ultrapassar o obstáculo.

1.3 Objetivos principais

O objetivo geral desta dissertação é a implementação de um algoritmo capaz de proporcionar a uma equipa de robôs terrestres a capacidade de navegação em formação até um destino pretendido evitando colisões, quer entre obstáculos quer entre robôs pertencentes à mesma equipa. Para o efeito, pretende-se fazer uso de uma abordagem onde a troca de informação através da comunicação sem fios entre robôs seja minimizada.

Os objetivos da dissertação são:

- Estudo de abordagens já realizadas para a navegação e estudo de técnicas de minimização da comunicação entre robôs;
- De entre as abordagens que se encaixam mais no problema que é proposto, analisar em detalhe as suas características para enunciar uma possível solução;

- Estudo e treino do software de simulação assim como do ambiente ROS;
- Desenvolvimento do algoritmo adotado e sua implementação em ROS;
- Realização de testes em ambiente de simulação com o algoritmo adotado para controlo de formação utilizando o simulador *Stage* (simulador baseado em ROS), com níveis de complexidade crescente:
 1. A equipa de robôs é posicionada com uma forma geométrica diferente da desejada, devendo convergir para esta formatura com o decorrer da movimentação. Neste teste não é necessário a existência de obstáculos;
 2. A equipa de robôs deve contornar obstáculos estáticos;
 3. A equipa de robôs deve deslocar-se em ambientes obstruídos;
 4. Os robôs devem ser dotados da capacidade de evitar colisões com obstáculos dinâmicos, ou seja, os robôs não podem em momento algum chocarem entre si;
 5. Validar o sistema proposto para formações de grande dimensão.
- Validação do algoritmo adotado para controlo de formação e suas diferentes variantes com robôs reais.

1.4 Estrutura do Documento

O documento está organizado em 6 capítulos:

- O capítulo 1 introduz o tema proposto, as motivações e os objetivos a serem alcançados;
- O capítulo 2 descreve o estado da arte. Menciona para cada tipo de estratégia de controlo de formação os trabalhos mais relevantes que têm sido apresentados;
- O capítulo 3 apresenta os conceitos chave das dinâmicas de atração aplicadas ao movimento de um robô;
- O capítulo 4 começa por expor as noções do ambiente ROS, apresenta plataformas robóticas utilizadas, os pressupostos do ambiente de simulação desenvolvido e termina por mostrar as dinâmicas de atração que podem ser aplicadas ao controlo de formação, tendo em conta as limitações do *hardware* utilizado;
- O capítulo 5 expõe o conjunto de experiências realizadas para comprovar o funcionamento da estratégia de controlo de formação implementada;
- O capítulo 6 apresenta a conclusão e propõe referências para trabalho futuro.

2 Estratégias de Controle de Formação

Na literatura encontram-se diversas estratégias possíveis para garantir que um grupo de robôs móveis se mantenha com uma forma geométrica desejada. Perante as estratégias existentes, existem algumas que são motivo de maior atenção nomeadamente a abordagem Baseada em Comportamentos, Líder-Seguidor, Estruturas Virtuais, Funções de Potencial e Teoria dos Grafos. Neste capítulo são apresentadas estas estratégias.

2.1 Abordagem Baseada em Comportamentos

Alguns dos comportamentos típicos de um robô numa abordagem baseada em comportamentos podem ser a tarefa de evitar obstáculos, deslocamento até à posição desejada ou a tarefa de garantir a sua permanência na formação. Estes comportamentos são, por conseguinte, colocados em campos de potencial, o que permite controlar o movimento da plataforma tendo em conta todas as contribuições existentes. A contribuição de cada comportamento varia com o ambiente que rodeia o robô, através dos dados recolhidos dos equipamentos sensoriais, como a posição dos obstáculos, a posição de robôs, a velocidade atual, a velocidade de outros robôs, ou até o estado de outro comportamento.

Em determinados projetos é vantajoso que exista um sistema de coordenação de comportamentos, por exemplo com a finalidade de escolher os comportamentos que influenciam a saída do sistema, como os comandos de velocidade a aplicar às rodas.

Balch e Arkin [3] apresentaram um esquema para o controlo da navegação em formação onde se consideram 5 parâmetros de entrada cada um com uma significância diferente. Estes parâmetros correspondem a sub-esquemas, com a finalidade de evitar um obstáculo estático, evitar robôs, movimentação até à posição objetivo, manter a forma geométrica e em simulação existe também um esquema para criar ruído com a finalidade do robô evitar máximos e mínimos locais. Com este tipo de comportamentos cada robô ganha a capacidade de se deslocar para a posição desejada ao mesmo tempo que evita obstáculos estáticos, evita colisões

com robôs e permanece em formação. Cada um dos parâmetros cria um vetor que representa o comportamento desejado, e com a soma de todos os vetores alcança-se o comportamento a aplicar no robô.

Naffin e Sukhatme [30], utilizaram uma estratégia baseada em comportamentos para implementar um mecanismo de socialização de robôs, onde qualquer robô pode ter um comportamento de solteiro, seguidor ou líder. Esta estratégia utiliza também uma computação distribuída e utiliza um sistema de localização local.

Monteiro e Bicho [27], aplicam a cada robô um sistema dinâmico não linear com a finalidade de controlar a orientação e um sistema dinâmico linear com o objetivo de controlar a velocidade de deslocação de cada robô, que contemplam para cada robô as tarefas de evitar obstáculos e a tarefa de convergir para a posição desejada na formação. A posição desejada na formação é projetada com o uso de uma estratégia líder-seguidor indicando-se para cada robô seguidor qual é o seu líder e a respectiva distância e orientação relativa desejada.

Como referido por Guanghai *et al.* [19], as abordagens baseadas em comportamentos permitem a utilização de um sistema paralelo, distribuído e de tempo real, reduzindo desta forma o fluxo informação necessário entre os robôs. Este tipo de abordagem é útil por exemplo no transporte de uma equipa de robôs num ambiente dinâmico e em constante alteração.

2.2 Abordagem Líder-Seguidor

Numa abordagem líder-seguidor, parte-se do princípio que, a cada robô, pode ser atribuída a etiqueta de líder ou de seguidor, onde o seguidor tem como principal função seguir o líder em função da distância e orientação relativas.

Uma das técnicas líder-seguidor mais populares foi apresentada por Desai, Ostrowski e Kumar [10], onde se utiliza a informação proveniente apenas de sensores locais, isto é sensores usados nos robôs móveis. Esta arquitetura contempla dois tipos de controlos com a finalidade de manter um grupo de robôs em formação, o controlador $l-\varphi$ e controlador $l-l$, cuja nomenclatura está apresentada na figura 2.1.

O objetivo do controlador $l-\varphi$ consiste em manter a distância relativa desejada l_{12}^d e o ângulo relativo desejado φ_{12}^d entre o líder e o seguidor. A lei de controlo faz a linearização das entradas e saídas de modo a que os valores das variáveis l_{12}^d e φ_{12}^d convirjam de forma exponencial para os valores desejados.

O controlador $l-l$ considera a posição relativa entre três robôs, e é atribuído a um dos

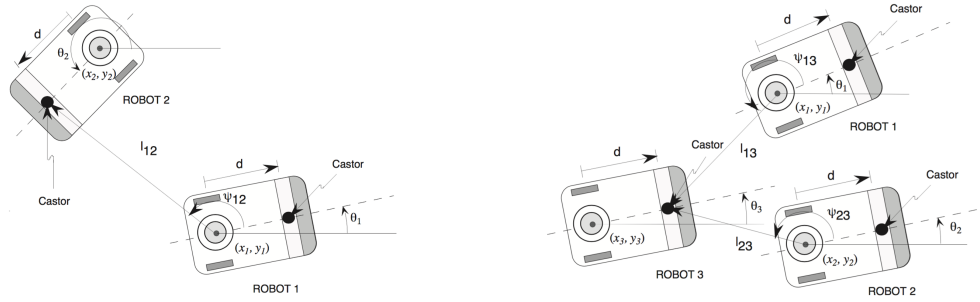


Figura 2.1: Notação dos controladores $l-\varphi$ (à esquerda) e $l-l$ (à direita), proposto por Desai, Ostrowski e Kumar [10].

robôs a função de seguir os outros dois. Esta função consiste em assegurar as distâncias relativas l_{13}^d e l_{23}^d , entre o seguidor e os dois líderes.

Neste tipo de controladores é necessário que o robô seguidor tenha conhecimento permanente tanto da distância e do ângulo relativo entre o robô líder e seguidor, como também das velocidades que o robô líder está a praticar. Só deste modo é possível manter a forma geométrica desejada.

Chetty *et al.* [8], apresentaram uma solução para o controlo de formação constituída por duas camadas, uma com a finalidade de navegação em formação e a outra com a finalidade de evitar obstáculos. Esta solução exige um sistema de posicionamento global, onde cada robô calcula a sua pose tendo por base as informações provenientes de *encoders*. Neste método o robô líder comunica a sua pose e respetivas velocidades para o robô seguidor através de um protocolo de comunicação TCP/IP numa rede sem fios.

Recentemente Cruz-Morales *et al.* [9], apresentaram uma solução para robôs holonómico, que determina a distância e orientação relativa entre o líder e o seguidor em tempo-real utilizando um sistema de localização baseado em visão montado a bordo do robô seguidor. Este sistema de localização consiste na utilização de um câmara Kinect, e recorre às funcionalidades existentes no *software* OpenCV, para uma informação mais detalhada consulte [7], com o intuito de reconhecer o marcador colocado a bordo da plataforma do líder e por conseguinte estimar a respetiva posição relativa.

2.3 Abordagem baseada em Estruturas Virtuais

A abordagem baseada em estruturas virtuais é muito utilizada em tarefas com veículos aéreos não tripulados, como satélites ou naves espaciais.

Esta estratégia foi proposta por Tan e Lewis [40], que definem uma estrutura virtual

como sendo uma coleção de elementos, como robôs, que mantêm uma forma geométrica (não obrigatoriamente rígida) entre si, respeitando a posição e orientação de um referencial pertencente à estrutura virtual.

O algoritmo proposto é constituído por 5 etapas: 1 - Alinhar a estrutura virtual tendo em conta a posição dos robôs, minimizando o erro de posição entre a estrutura virtual e a posição dos robôs; 2 - Movimentar a estrutura virtual com uma distância Δx e orientação $\Delta \theta$, determinada pela trajetória desejada para a equipa de robôs; 3 - Determinar a trajetória desejada para cada robô de forma a convergir para a sua respetiva posição na estrutura virtual; 4 - Ajustar os comandos de velocidade de cada robô para seguir a trajetória desejada; 5 - Regressar à etapa 1.

Ou seja, nas abordagens baseadas em estruturas virtuais considera-se a equipa de robôs como um corpo rígido, onde cada robô tem como finalidade deslocar-se em função do deslocamento da estrutura virtual.

Recentemente Ogunbodede e Huijberts [31] apresentaram dois algoritmos de controlo de formação para robôs aéreos não tripulados, um algoritmo hierárquico e um algoritmo não hierárquico. Ambos consideram a existência de um líder virtual no centro da formação e em função da sua posição é determinada a posição desejada para a equipa de robôs.

O algoritmo hierárquico considera, para além do líder virtual, a existência de um líder secundário para cada robô, recorrendo à estratégia baseada na teoria dos grafos, onde o robô secundário corresponde a um dos robôs da equipa. Em função das posições dos 2 líderes é determinada a posição que o robô deve assumir, o que garante uma permanência rígida da forma geométrica desejada.

Por outro lado o algoritmo não hierárquico baseia-se apenas na posição do líder virtual para determinar a posição que cada robô deve tomar, garantindo assim a forma geométrica desejada. O facto de não utilizar um líder secundário conduz a uma solução com maiores erros de posição. Porém, é uma solução que exige menos recursos computacionais e a movimentação de cada robô é conseguida de forma suave.

2.4 Funções de Potencial Artificial

O conceito de potencial artificial foi proposto por Khatib [23], que considera que os robôs movem-se usando um campo de forças onde a posição objetivo representa um pólo de atração, originado pela função de potencial atrativa, e os obstáculos são superfícies repulsivas, criados pela função de potencial repulsiva. Por cada instante de tempo somam-se as contribuições

resultantes do cálculo do gradiente negativo das funções de potencial atrativas e repulsivas, evitando-se assim colisões entre robôs e obstáculos, ao mesmo tempo que é garantida a convergência do robô para a posição objetivo.

Yao, Ordóñez e Gazi [47], utilizaram como base funções de potencial artificial para realizar o controlo de formação. Neste caso cada robô inclui duas funções de potencial atrativas, onde numa função de potencial existe o termo de seguimento e na outra o termo de manter a formação. Com estas duas funções quando um robô se desloca de forma diferente do resto do grupo, o grupo reage a este comportamento deslocando-se para perto dele.

Sabattini *et al.* [39] apresentam uma estratégia de controlo baseada na interação de campos de potencial com a finalidade de criar formas geométricas arbitrárias.

Bennet *et al.* [4] aplicam a teoria da bifurcação em campos de potencial com a finalidade de reconfigurar padrões geométricos apenas com a alteração de um parâmetro.

A abordagem apresentada por Tsankova e Isapov [41], considera funções de potencial para a distância entre robôs, a orientação da formação, o comportamento de evitar obstáculos e também para a deslocação até à posição objetivo na formação.

2.5 Abordagem baseada na Teoria dos Grafos




Na abordagem baseada na teoria dos grafos aplicada ao controlo de formação tipicamente os nós representam robôs e as arestas que ligam os nós representam constrangimentos existentes entre os robôs.

Um dos trabalhos pioneiros foi proposto por Olfati-Saber e Murray [32], que definem um conjunto de noções úteis como a influência da rigidez do grafo, as formas de representação formal para um grupo de robôs (divisão, reunião e manobras de reconfiguração) e uma formulação algébrica para os grafos.

Fax e Murray [12] utilizaram grafos para representar as tipologias de comunicação entre robôs. Contudo, a estabilidade da formação é alcançada apenas quando não existem falhas no fluxo de informação, tornando o sistema sensível a erros sensoriais.

Recentemente Montijano [29] *et al.* propuseram uma abordagem distribuída para o controlo de formação de uma equipa de quadricópteros, onde faz uso de um sistema de posicionamento relativo baseado em visão por computador, que conjuga as imagens do solo capturadas pelas câmaras dos quadricópteros.

2.6 Comparação entre Estratégias de Controlo de Formação

Na tabela 2.1 indica-se para cada abordagem as características que são verificadas , as não verificadas  e as que podem vir a ser verificadas com a conjugação de outro método .



















	Baseada em Comportamentos	Líder-Seguidor	Estruturas Virtuais	Funções de Potencial Artificial	Teoria dos Grafos
Criar Forma Geométrica					
Reconfiguração e Seleção da Forma					
Acompanhamento da Formação					

Tabela 2.1: Comparação entre estratégias de controlo de formação, características verificadas, não verificadas e possivelmente verificadas, proposto por Guanhua *et al.* [19].

Na lista de características apresentada na tabela 2.2, está presente uma avaliação para cada abordagem. A avaliação diferencia-se por capacidade elevada , capacidade média  ou capacidade baixa .































	Baseada em Comportamentos	Líder-Seguidor	Estruturas Virtuais	Funções de Potencial Artificial	Teoria dos Grafos
Variedade de Formações					
Precisão da Formação					
Estabilidade da Formação					
Flexibilidade ou Adaptabilidade					
Solução de Tempo-Real					
Aplicação Distribuída					

Tabela 2.2: Comparação entre estratégias de controlo de formação, capacidade de execução das características, proposto por Guanhua *et al.* [19].

2.7 Sumário

Neste capítulo, foi feita uma introdução das abordagens mais relevantes que se encontram na literatura. Para cada abordagem apresentam-se exemplos de trabalhos pioneiros e de trabalhos estudados recentemente. Com o estudo das abordagens existentes optou-se por escolher uma abordagem Baseada em Comportamentos, que será exibida no próximo capítulo.

3 Dinâmicas de Atração

Neste capítulo introduz-se as dinâmicas de atração com o objetivo posterior de criar comportamentos para as plataformas robóticas usadas neste trabalho, tendo como base o trabalho desenvolvido por Bicho [5] e Monteiro [28]. Começa-se por apresentar os princípios básicos das dinâmicas de atração e finaliza-se com a transposição destas dinâmicas para plataformas robóticas onde os objetivos são a deslocação em ambientes desconhecidos e até um local objetivo.

3.1 Princípios Básicos das Dinâmicas de Atração

Para se desenhar uma arquitetura baseada numa abordagem de dinâmicas de atração tem de se escolher as variáveis ambientais, que segundo Bicho [5] *têm a finalidade de descrever, quantificar e representar internamente o estado do sistema.*

No trabalho desenvolvido os robôs deslocam-se num plano, e deste modo a orientação ϕ do robô relativamente a um eixo de coordenadas fictício e comum a todos os robôs e a velocidade de deslocação v , são variáveis ambientais adequadas ao problema.

Os comandos de velocidade gerados pelas soluções dos sistemas dinâmicos ou dinâmicas ambientais alteram o comportamento do robô, uma vez que têm como finalidade modificar o estado das variáveis ambientais. Por conseguinte, os sistemas dinâmicos são formados pelas equações diferenciais 3.1 e 3.2.

$$\dot{\phi} = f(\phi, \text{parâmetros}) \quad (3.1)$$

$$\dot{v} = g(v, \text{parâmetros}) \quad (3.2)$$

Quando a variável ambiental não sofre alterações, situação que ocorre quando as soluções das funções diferenciais são zero, significa que o sistema dinâmico se encontra numa solução constante podendo também ser chamada de ponto fixo.

A estabilidade dos pontos fixos depende da taxa de variação dos mesmos, no caso de se tratar de uma variação positiva então o sistema dinâmico sofre repulsão para fora do ponto fixo o que significa que se trata de um ponto fixo instável, ou seja de uma dinâmica de repulsão como apresentado na figura 3.1a. Na situação inversa quando se trata de uma variação negativa o sistema dinâmico é atraído para o ponto fixo sendo este um ponto fixo assintoticamente estável, tratando-se de uma dinâmica de atração como apresentado na figura 3.1b.

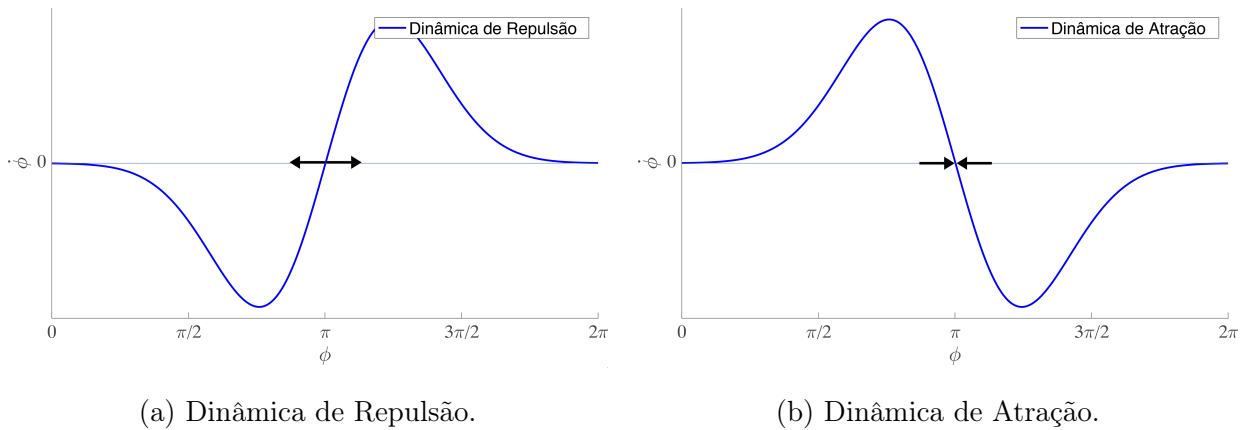


Figura 3.1: Pontos fixos dos sistemas dinâmicos.

O meio que rodeia o robô modela o seu comportamento, com a inclusão de forças atrativas e repulsivas. No caso de se tratar de uma força atrativa, a distância e a direção a que se encontra o líder relativamente à posição do seguidor gera uma velocidade linear e angular desejada. No caso de se tratarem de obstáculos, dependendo da proximidade do obstáculo ao robô e da orientação a que se encontra, a força gerada por este obstáculo é diferente. Por exemplo, quando o obstáculo se encontra à frente do robô a sua significância deve ser maior.

Estas forças por sua vez criam uma contribuição para um estado instável ou um estado assintoticamente estável. No entanto, não deixa de se tratar de um sistema dinâmico não linear pois o meio que envolve os robôs varia gerando muitas contribuições atrativas e repulsivas, que não permitem manter a estabilidade total do sistema.

3.2 Aplicação das Dinâmicas de Atração para o Movimento de um Robô

Com as variáveis ambientais escolhidas, ϕ e v , resta determinar o modo de atribuição de valores para as dinâmicas ambientais. Para isso, apresentam-se de seguida os princípios

usados para atribuir valores a $\dot{\phi}$ e \dot{v} , tendo em consideração tarefas como deslocamento do robô até um local objetivo e a tarefa de evitar obstáculos.

3.2.1 Orientação do Robô

Os valores desejados para a orientação do robô ϕ dependem do ambiente que o rodeia. Na figura 3.2 apresenta-se um ambiente que um robô tipicamente encontra, porém com uma simples alteração no ambiente, contribuições diferentes são criadas.

No caso ilustrativo, figura 3.2, a orientação do robô é alterada em função de ψ_{obj} e $\psi_{obs,l}$, que representam o local objetivo e a contribuição do obstáculo representada numa leitura do LRF que detetou o obstáculo.

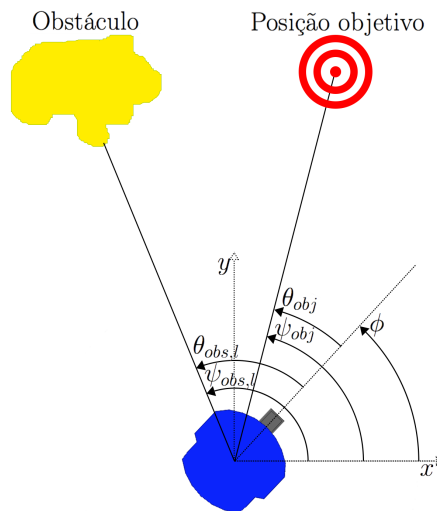


Figura 3.2: Demonstração do ambiente que rodeia o robô.

Na tarefa de evitar obstáculos a disposição dos obstáculos estáticos ou dinâmicos representam direções indesejadas para o robô, pelo que se refletem numa força repulsiva. Com o uso de múltiplos robôs é frequente que numa equipa os robôs representem também obstáculos a evitar, claro está que o líder não representa um obstáculo.

Dependendo do ambiente que rodeia o robô recorrendo à informação proveniente de um sensor LRF, para cada leitura l fornecida pelo sensor, com $l = 1, 2, 3, \dots, N$ onde N é o número total de leituras facultadas pelo LRF, é gerada uma força repulsiva, equação 3.3.

$$f_{obs,l}(\phi) = \lambda_{obs,l}(\phi - \psi_{obs,l})e^{-\frac{(\phi - \psi_{obs,l})^2}{2\sigma_{obs,l}^2}} \quad (3.3)$$

Com uso de um LRF em cada robô, para cada leitura proveniente do mesmo é possível determinar a distância a que foi detetado o obstáculo $d_{obs,l}$ e a orientação $\psi_{obs,l}$ relativamente ao centro de coordenadas do robô. Como $\phi - \psi_{obs,l} = -\theta_{obs,l}$, significa que não é necessário

o conhecimento da orientação do robô num sistema de coordenadas global ϕ nem mesmo a orientação $\psi_{obs,l}$, mas sim apenas a posição angular da leitura do LRF nas coordenadas do robô para calcular cada força repulsiva.

A força $\lambda_{obs,l}$, equação 3.4, varia com a distância $d_{obs,l}$, onde uma menor distância corresponde a uma maior força de repulsão, e os parâmetros β_1 e β_2 controlam a força global e a taxa de decaimento, respetivamente.

$$\lambda_{obs,l} = \beta_1 e^{-\frac{d_{obs,l}}{\beta_2}} \quad (3.4)$$

A variável $\sigma_{obs,l}$ representa o intervalo angular, equação 3.5, onde a força repulsiva para a leitura l atua, que depende do setor que a leitura abrange $\Delta\zeta$, da distância a que o obstáculo foi sentido $d_{obs,l}$, e também do raio do robô R_{robo} .

$$\sigma_{obs,l} = \arctan \left[\tan \left(\frac{\Delta\zeta}{2} \right) + \frac{R_{robo}}{R_{robo} + d_{obs,l}} \right] \quad (3.5)$$

Com a inclusão do raio do robô no cálculo de $\sigma_{obs,l}$, aumenta o valor de $\sigma_{obs,l}$ o que garante que o robô tem a possibilidade de passar ao lado do obstáculo. Este acréscimo representa um ângulo de segurança.

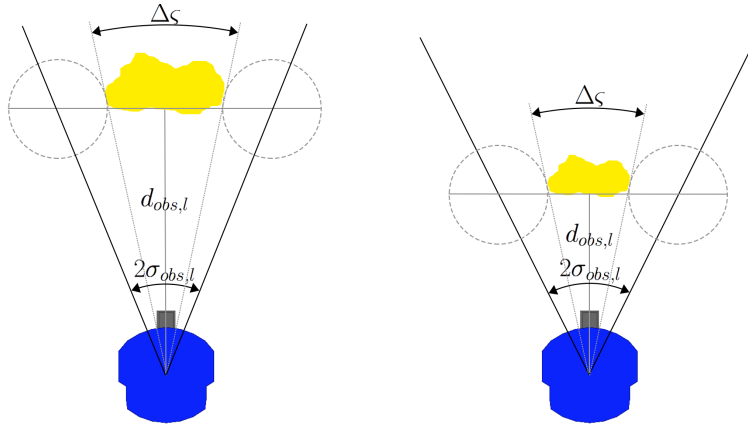


Figura 3.3: Intervalo angular onde a força repulsiva atua.

Pela figura 3.3, é perceptível que quando o obstáculo se situa mais perto do robô, o ângulo de segurança é maior, de modo a garantir a passagem do robô sem chocar com o obstáculo.

A figura 3.4 mostra uma resposta típica da força repulsiva para uma leitura proveniente do LRF.

Com o aumento do número de leituras do LRF disponíveis, a força de cada leitura β_1 tem de sofrer uma redução para que quando se realizar o somatório de todas as forças repulsivas esta soma não se reflita em valores constantemente elevados.

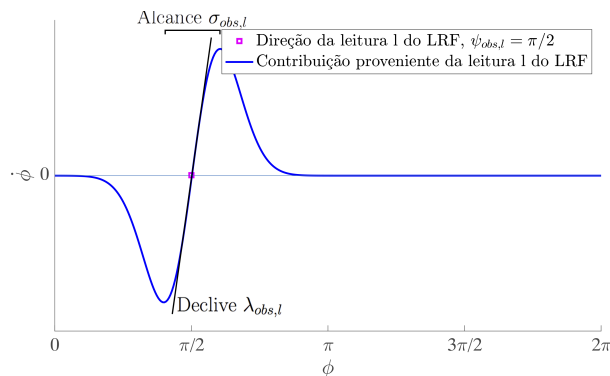
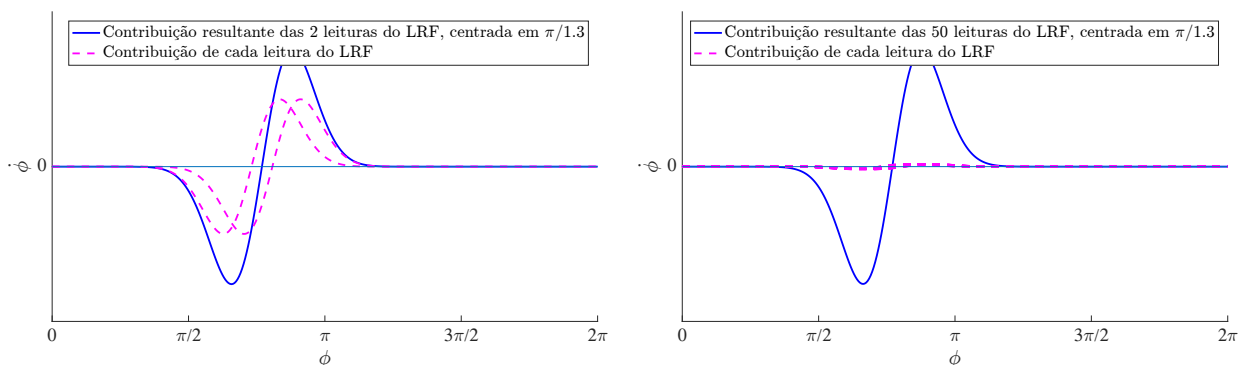


Figura 3.4: Forma da resposta repulsiva.

A dinâmica de evitar obstáculos para a orientação do robô é dada pelo somatório de todas as contribuições provenientes das leituras do LRF, como apresentado na equação 3.6.

$$\dot{\phi} = F_{obs}(\phi) = \sum_{l=1}^N f_{obs,l}(\phi) \quad (3.6)$$

Pelas figuras 3.5, demonstra-se o efeito do aumento da resolução angular, onde a figura 3.5a apresenta um resultado com menor resolução que o resultado presente na figura 3.5b. De tal modo que para detetar um determinado obstáculo com o sensor de menor resolução apenas 2 leituras correspondem à posição do obstáculo, já no sensor de maior resolução 50 leituras contêm informação sobre o obstáculo.



(a) Contribuição de 2 leituras.

(b) Contribuição de 50 leituras.

Figura 3.5: Comparação da contribuição entre sensores de baixa e de alta resolução.

Como já referido anteriormente, com o aumento do número de leituras disponíveis num LRF para a mesma área de abrangência, aumentando a resolução do sensor, a força singular de cada leitura deve ser reduzida. Como se pode verificar, na figura 3.5b a força contributiva de cada leitura em comparação com a figura 3.5a é menor. No entanto, com um bom ajustamento da variável $\lambda_{obs,l}$ a força resultante para os dois sensores pode encontrar-se na mesma gama.

Em função da posição objetivo é gerada uma força atrativa para esse local, equação 3.7, que é dependente da direção em que se encontra a posição objetivo, ψ_{obj} .

$$f_{obj}(\phi) = -\lambda_{obj} \sin(\phi - \psi_{obj}) \quad (3.7)$$

Com a força do tipo sinusoidal apresentada na equação 3.7, é criada uma força atrativa com um limiar máximo definido por λ_{obj} para a direção ψ_{obj} , e quando a orientação do robô está no lado oposto ao desejado, em $\psi_{obj} + \pi$, esta força atrativa reflete-se numa repulsão a essa orientação. Na figura 3.6 encontra-se uma resposta típica de uma contribuição atrativa.

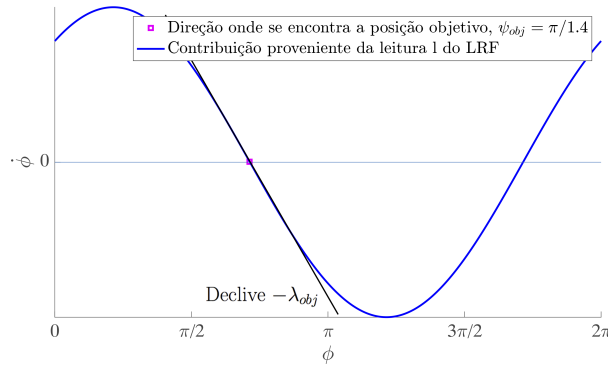


Figura 3.6: Forma da resposta atrativa.

Com o uso de um sensor que detete a direção em que se encontra a posição objetivo relativamente ao sistema de coordenadas do robô, deixa de ser necessário o conhecimento da variável ϕ que obriga a ter um referencial comum a todos os robôs. Neste trabalho fez-se uso de uma câmara a bordo de cada robô para detetar a direção a que se encontra o líder respetivo, direção esta atribuída a θ_{obj} , deste modo podemos reescrever $\phi - \psi_{obj} = -\theta_{obj}$.

O sistema dinâmico resultante para a orientação do robô, equação 3.8, engloba a contribuição de todos os obstáculos detetados pelo LRF e também a contribuição da posição objetivo no mesmo instante temporal.

$$\dot{\phi} = F_{obs}(\phi) + f_{obj}(\phi) + f_{estoc} \quad (3.8)$$

Como esta solução usa campos vectoriais é possível a ocorrência de mínimos locais, o que pode acontecer quando a força de atração anula a força de repulsão, originando-se assim uma força nula colocando o robô num ponto fixo instável. Para garantir que o robô não fica estagnado nestes pontos, faz-se uso de uma função estocástica, f_{estoc} .

No presente sistema optou-se por atribuir no instante em que ocorre um mínimo local a contribuição $f_{estoc} = f_{obj}(\phi)$, permitindo assim que a plataforma robótica saia da orientação

de bloqueio em que se encontra, minimizando ao mesmo tempo o erro de orientação. Desta forma evita-se que a contribuição atribuída a f_{estoc} ocorra na direção contrária à que se encontra a posição objetivo, o que mais tarde pode fazer com que o robô convirja novamente para o mesmo mínimo local.

A contribuição dos obstáculos mencionada na figura 3.5b e a contribuição da posição objetivo mencionada na figura 3.6, resulta no sistema dinâmico apresentado na figura 3.7.

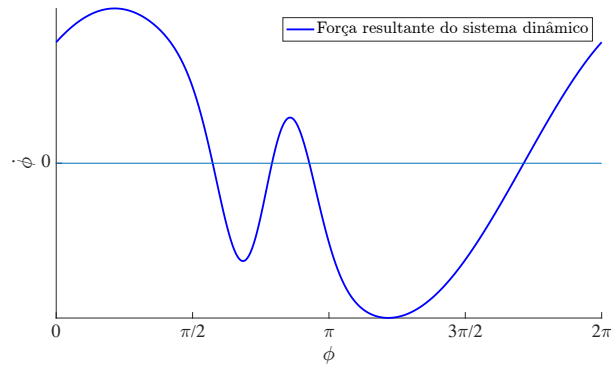


Figura 3.7: Forma da resposta dinâmica para a orientação do robô.

Embora o LRF contenha N leituras disponibilizadas, nem todas correspondem à detecção de obstáculos, visto que parte das leituras podem pertencer à posição do robô líder ou até mesmo a erros de leituras, tendo estas que ser ignoradas. Trata-se de um sistema dinâmico não linear, pois existe um número incerto de forças atrativas e repulsivas.

3.2.2 Velocidade de Deslocação

O controlo de velocidade linear é fulcral para garantir a permanência do robô na posição desejada ou perto desta, mas também para assegurar que o robô não colida com obstáculos.

O sistema dinâmico usado para o controlo da velocidade de deslocação é dado pela equação 3.9.

$$\dot{v} = \gamma_{obs}g_{obs}(v) + \gamma_{obj}g_{obj}(v) \quad (3.9)$$

Onde as contribuições $g_{obs}(v)$ e $g_{obj}(v)$ correspondem a sistemas dinâmicos lineares, dadas pelas equações 3.10 e 3.11, que são ativadas por intermédio das variáveis mutuamente exclusivas γ_{obs} e γ_{obj} . Estas variáveis mutuamente exclusivas são ativadas e desativadas dependendo se existem obstáculos na frente do robô que coloquem perigo de colisão. No caso de existir perigo de colisão a variável γ_{obs} toma o valor 1 e a variável γ_{obj} toma o valor 0. Desta forma apenas o sistema dinâmico $g_{obs}(v)$ fornece valores para \dot{v} . No caso de não existir

perigo de colisão a variável γ_{obs} toma o valor 0 e a variável γ_{obj} toma o valor 1, o que significa que os valores atribuídos a \dot{v} são provenientes do sistema dinâmico $g_{obj}(v)$.

$$g_{obs}(v) = -\alpha_{obs}(v - v_{d,obs}) \quad (3.10)$$

$$g_{obj}(v) = -\alpha_{obj}(v - v_{d,obj}) \quad (3.11)$$

Os sistemas dinâmicos lineares apresentados anteriormente, representam forças atrativas para as velocidades desejadas $v_{d,obs}$ e $v_{d,obj}$, com uma taxa de relaxamento de $\alpha_{obs} > 0$ e $\alpha_{obj} > 0$, respetivamente.

A velocidade desejada na existência de perigo de colisão está contida na variável $v_{d,obs}$, equação 3.12, que é a função da distância mínima medida pelo LRF d_{min} , e do tempo pretendido até ocorrer contacto com o obstáculo $T_{2c,obs}$.

$$v_{d,obs} = \frac{d_{min}}{T_{2c,obs}} \quad (3.12)$$

Quando não existe perigo de colisão a velocidade desejada é dada por $v_{d,obj}$, equação 3.13, que está em função da distância a que se encontra a posição objetivo d_{obj} , e do tempo pretendido até ser estabelecido contacto com a posição desejada $T_{2c,obj}$. Uma vez que a velocidade do robô cresce com a distância a que se encontra a posição objetivo, é conveniente limitar a velocidade a um valor, v_{max} .

$$v_{d,obj} = \begin{cases} \frac{d_{obj}}{T_{2c,obj}} & \text{se } v_{d,obj} \leq v_{max} \\ v_{max} & \text{caso contrário} \end{cases} \quad (3.13)$$

A função de potencial apresentada em 3.14 tem o objetivo determinar se existem obstáculos à frente do robô que criem perigo de colisão.

$$U_{obs}(\phi) = \sum_{l=1}^N \lambda_{obs,l} \sigma_{obs,l}^2 \left(e^{-\frac{(\phi - \psi_{obs,l})^2}{2\sigma_{obs,l}^2}} - \frac{1}{\sqrt{e}} \right) \quad (3.14)$$

A figura 3.8 apresenta resposta típica da função de potencial. É de salientar que para os valores positivos da função de potencial, a orientação do robô encontra-se suficientemente numa região repulsiva, o que significa que existe perigo de colidir com um obstáculo. Com o fim de indicar se a variável U_{obs} contém valores positivos ou negativos, utilizou-se a variável

booleana U_{pot} que toma valores como indicado na equação 3.15.

$$U_{pot} = \begin{cases} 1 & \text{se } U_{obs} \geq 0 \\ 0 & \text{caso contrário} \end{cases} \quad (3.15)$$

No caso de $U_{pot} = 1$ então $\gamma_{obs} = 1$ e $\gamma_{obj} = 0$, na situação inversa se $U_{pot} = 0$ a variável $\gamma_{obs} = 0$ e $\gamma_{obj} = 1$, o que permite controlar o sistema dinâmico para a velocidade de deslocação, equação 3.9.

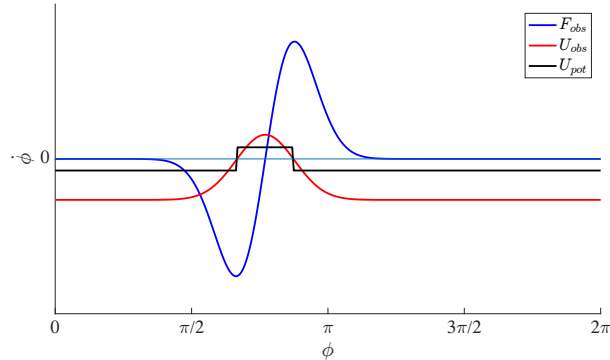


Figura 3.8: Exemplo da função potencial da orientação do robô para evitar obstáculos.

3.3 Sumário

Este capítulo apresentou os princípios das dinâmicas de atração, indicando as variáveis ambientais usadas e os respectivos sistemas dinâmicos com a finalidade de alterar o estado das mesmas. Estes sistemas dinâmicos abrangem em simultâneo, tanto a capacidade do robô móvel alcançar a posição objetivo como a capacidade de evitar obstáculos.

O controlo de formação, apresentado no próximo capítulo, é alcançado com o uso das dinâmicas de atração exibidas.

4 Sistema Desenvolvido

Neste capítulo apresenta-se em primeiro lugar as características *hardware* utilizado nas experiências, nomeadamente as características das plataformas robóticas e dos equipamentos externos à plataforma afim de realizar o reconhecimento ambiental como obstáculos e detectar o respetivo robô líder. Posteriormente indicam-se pressupostos e convenções utilizados no desenvolvimento do simulador, e finalmente o capítulo termina com a inclusão das dinâmicas de atração, tendo como base o trabalho desenvolvido por Monteiro [28], para a criação de formas geométricas do tipo coluna e oblíquo atendendo às restrições de *hardware* existentes.

4.1 Ambiente ROS

Com base no documento [34] é apresentada a nomenclatura utilizada na implementação de programas baseados em ROS.

4.1.1 Nós

A arquitetura ROS foi desenhada para ter processos modulares, onde cada módulo é responsável por desempenhar uma tarefa computacional. A estes módulos chama-se de **nós**.

No pacote **robot_formation**, desenvolvido no trabalho de dissertação proposto, existe um conjunto de nós que têm diferentes tarefas a realizar. Por exemplo, um dos nós existente é o **rf_node_column_follower** que tem como objetivo garantir a convergência da plataforma para a posição desejada numa forma geométrica do tipo coluna em ambiente real.

4.1.2 Mensagens

Os nós podem comunicar entre si com recurso a **mensagens**, mensagens estas do tipo estrutura que podem ser constituídas por inteiros, vírgulas flutuantes, booleanos ou até mesmo ser compostas por outras mensagens.

No caso do nó `rf_node_column_follower`, este subscrive às mensagens do tipo `sensor_msgs` com a finalidade de recolher a informação proveniente do LRF. A informação contida neste tipo de mensagens está descrita em [36].

4.1.3 Tópicos

Os **tópicos** são identificados pelo seu nome do tipo *string* e representam o endereço do local onde estão armazenadas mensagens. Um nó pode aceder a estes tópicos tanto para publicar mensagens como para subscriver a dados contidos na mensagem.

Um tópico pode ser subscrito ou pode ser publicado por diversos nós, e um nó pode subscriver ou publicar em diversos tópicos.

Este tipo de trocas de mensagens são do tipo *broadcast*, onde os nós estão constantemente à espera de novas atualizações dos tópicos.

Recorrendo ao exemplo apresentado anteriormente, as mensagens do tipo `sensor_msgs` são publicadas pelo nó `hokuyo_node` no tópico `/scan` e são subscritas pelo nó `rf_node_column_follower`.

4.1.4 Serviços

Um **serviço** em ROS é composto por um nome que o identifica e por um par de mensagens: uma mensagem com a finalidade de pedir o serviço e outra mensagem para guardar a resposta.

Utilizando o mesmo exemplo, após a recolha da informação contida numa mensagem do tipo `sensor_msgs`, o nó `rf_node_column_follower` precisa de utilizar o serviço `lookupTransform` com a finalidade de transformar todas as leituras do LRF para o eixo de coordenadas do robô, visto que as leituras são relativas do eixo de coordenada do LRF. Para uma consulta detalhada acerca do serviço consulte [16].

4.2 Plataformas Robóticas

Neste trabalho utiliza-se as plataformas robóticas de condução diferencial Pioneer 3-DX, que de acordo com Omron Adept [1] é a plataforma robótica de investigação mais popular.

A plataforma suporta o transporte de uma massa até 17 Kg o que torna possível adicionar equipamentos sensoriais e de controlo. Por exemplo, cada plataforma utilizada neste trabalho tem instalado um sensor LRF na parte dianteira com a finalidade de detetar obstáculos, e transporta também um computador para realizar tarefas de controlo em ambiente ROS.

Cada computador utilizado contém uma câmara integrada que permite a deteção de robôs da mesma equipa. Na figura 4.1 encontra-se discriminado um exemplar utilizado.

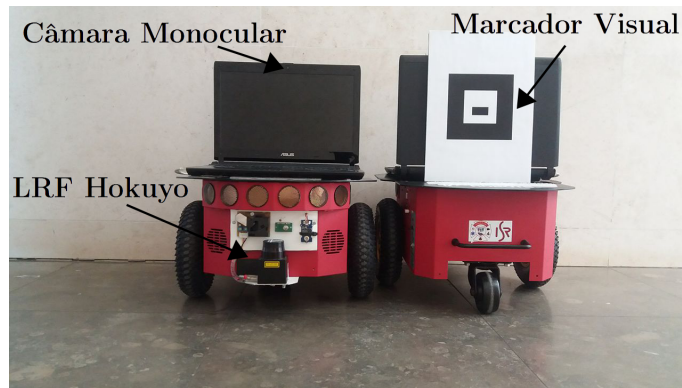


Figura 4.1: Plataforma robótica utilizada.

Como a plataforma contém *encoders* no eixo das rodas motrizes e a arquitetura de controlo foi implementada em ambiente ROS, é possível determinar a velocidade corrente da plataforma subscrevendo a informação contida nas mensagens de odometria, como sugerido em [38], tornando assim possível obter a velocidade linear da plataforma v_i .

4.2.1 Determinação da Posição Relativa do Robô Líder

Para que o robô seguidor tenha a capacidade de detetar a posição do seu robô líder utilizou-se a câmara monocular que vem integrada com o computador instalado na plataforma.

Embora com a utilização de uma câmara surjam alguns constrangimentos, como a oclusão do robô líder causada pelas variações de luminosidade não controlada ou pela ocorrência de obstáculos entre os robôs, o facto de ser possível testar e comprovar o comportamento da arquitetura de controlo de formação e, ao mesmo tempo, não ser necessário despendere mais dinheiro em equipamento extra a colocar na plataforma robótica, tornou esta solução adequada.

Após se obter os parâmetros de calibração da câmara utilizando o pacote **camera_calibration**, desenvolvido por Bowman e Mihelich [6], ficou disponível a matriz de projecção que contém a matriz dos parâmetros intrínsecos da câmara, como referido em [45].

Pelos conceitos indicados por Ashwin em [2], a matriz dos parâmetros intrínsecos tem a forma $\begin{bmatrix} f_x & s & x \\ 0 & f_y & y \\ 0 & 0 & 1 \end{bmatrix}$, onde f_x e f_y são a distância focal nas direções x e y respetivamente, s é o

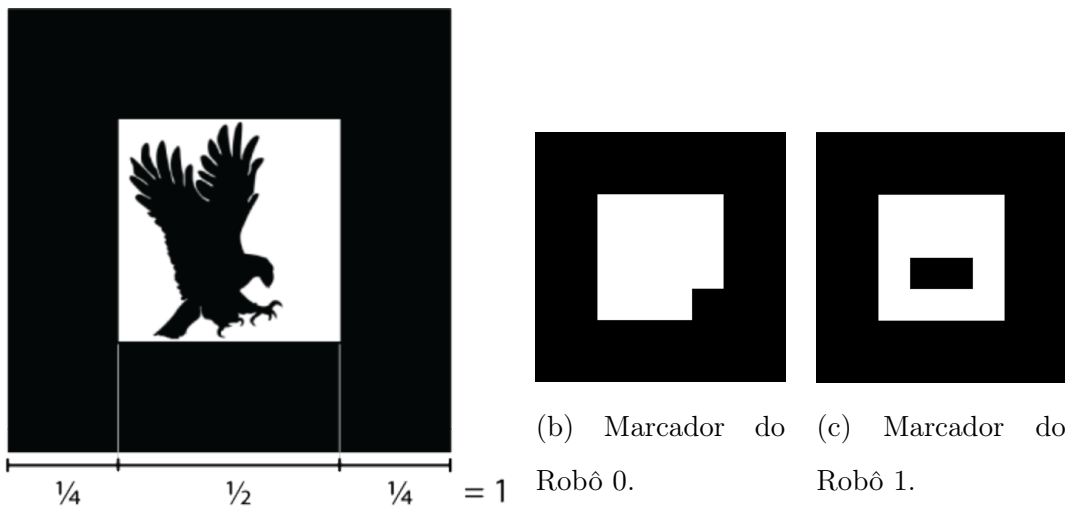
coeficiente de inclinação do eixo que tipicamente é 0, e x e y representam as coordenadas onde a imagem é produzida na câmara relativamente ao centro da imagem, sendo que, idealmente, estas coordenadas coincidem com o centro da imagem. De forma semelhante ao sugerido, o campo de visão da câmara pode ser dado pela equação 4.1.

$$a_x = \arctan\left(\frac{x}{f_x}\right) \quad (4.1)$$

Para as câmaras que incorporam os computadores instalados nas plataformas obteve-se um campo de visão entre -0.457 e 0.457 radianos, o que representa mais um constrangimento, para além dos indicados anteriormente comuns às maioria das soluções que utilizam visão por computador, pois limita o tipo de formas geométricas possíveis de realizar, sem que o líder saia do campo de visão limitado.

Com a finalidade de detetar a posição do robô líder relativamente ao robô seguidor, cada líder contém um marcador visual único e distinto de todos os outros existentes. Estes marcadores são colocados na parte traseira de cada computador, atrás da câmara e no centro da plataforma robótica, como demonstrado na figura 4.1.

Os marcadores visuais devem respeitar as dimensões definidas por ARToolworks Support Library em [24], dimensões estas apresentadas na figura 4.2a. Nas experiências em ambiente real, no capítulo 5, fez-se uso dos marcadores apresentados nas figuras 4.2b e 4.2c, possibilitando desta forma a identificação dos robôs líderes.



(a) Dimensões dos marcadores visuais, imagem retirada de [24].

Figura 4.2: Dimensões dos marcadores visuais e marcadores utilizados nas experiências em ambiente real.

Cada computador instalado nos robôs seguidores corre o pacote ROS **ar_pose**, desenvolvido por Dryanovsk, Morris e Dumonteil [11], com o objetivo de estimar a pose do eixo de coordenadas do marcador pertencente ao robô líder respectivo.

O pacote ROS **tf** desenvolvido por Foote, Marder-Eppstein e Meeussen que se encontra retratado em [15], permite a interligação de todos os eixos de coordenadas existentes e disponibiliza a capacidade de em qualquer momento relacionar dois eixos de coordenadas afim de saber qual a relação geométrica existente entre eles. Exemplos de eixos de coordenadas típicos são por exemplo a base do robô, a câmara do computador, o LRF e até o marcador visual. Estes autores desenvolveram também um conjunto de funções mencionadas em [16] para um fácil manuseamento com os eixos de coordenadas. Abaixo encontram-se exemplos de algumas funções utilizadas.

Por exemplo através do serviço **lookupTransform**, em qualquer momento é possível saber a relação geométrica existente entre o eixo de coordenadas do marcador visual e o eixo de coordenadas da câmara. Uma vez conhecida a relação geométrica, entre o marcador e a câmara, tem-se a distância a que se encontra o robô líder l_i , e também a orientação θ_i relativamente ao referencial do robô seguidor.

O serviço **lookupTwist**, permite determinar a velocidade linear atual do eixo de coordenadas do marcador $v_{marcador}$, mas a velocidade adquirida contém englobada a velocidade atual do robô seguidor v_i . Afim de eliminar esta contribuição aplica-se a equação 4.2, obtendo-se assim a velocidade real do robô líder v_j .

$$v_j = v_{marcador} - v_i \quad (4.2)$$

4.2.2 Detecção de Obstáculos

Durante a movimentação da plataforma, é necessário que esta tenha a capacidade de conhecer o ambiente que o rodeia pois existem obstáculos estáticos e dinâmicos que devem alterar a direção do robô de modo a evitar colisões entre a plataforma e os obstáculos. Daqui surge a necessidade de incorporar um LRF na parte dianteira da plataforma, como está explicito na figura 4.1. Assim os robôs passam a ter a capacidade de evitar tanto obstáculos estáticos como cadeiras ou paredes, mas também objetos dinâmicos como pessoas ou até mesmo os seus próprios robôs parceiros de equipa.

Porém, quando um determinado robô está a seguir um outro robô, esse robô denominado de líder não deve ser tratado como um obstáculo. Para tal todas as leituras do LRF que

correspondem à posição do robô líder são desconsideradas.

Para que as leituras do LRF correspondentes à posição do robô líder sejam desconsideradas, fez-se uso da equação 4.3 para se calcular a distância euclidiana entre cada leitura l do LRF, com $l = 1, 2, 3 \dots N$ onde N corresponde ao número de leituras facultadas pelo LRF, e a posição do robô nesse mesmo instante.

$$d_{LRF-robot,l} = \sqrt{d_{obs,l}^2 + l_i^2 + 2d_{obs,l}l_i \cos(\theta_{obs,l} - \theta_i)} \quad (4.3)$$

A variável $d_{obs,l}$ corresponde à distância do robô ao obstáculo e $\theta_{obs,l}$ a orientação da leitura no referencial do robô. Uma vez que o robô se assemelha a um cilindro, todas as leituras do LRF que se encontrem com uma distância maior que o raio do robô líder R_{robo} , isto é $d_{LRF-robot,l} > R_{robo}$, são consideradas como obstáculos, caso contrário desconsideram-se e não afetam o sistema dinâmico.

Neste trabalho fez-se uso de um LRF Hokuyo, detalhado em [21], e subscreve-se às mensagens do tipo `sensor_msgs` publicadas pelo nó `hokuyo_node` no tópico `/scan`, para uma informação mais detalhada acerca deste tipo de mensagens e deste tipo de nó consulte [36] e [18], respetivamente.

Como as mensagens do LRF são relativas ao eixo de coordenadas do LRF, para se obter a distância $d_{obs,l}$ e a orientação $\theta_{obs,l}$ é necessário fazer uma transformação das leituras para o eixo de coordenadas do robô, utilizando mais uma vez o serviço `lookupTransform`.

4.2.3 Aplicar velocidades à plataforma

Quando se tem as soluções do sistema dinâmico \dot{v} e $\dot{\phi}$, estas têm de ser aplicadas ao robô, para que as variáveis ambientais v e ϕ alterem o seu estado.

Em ambiente ROS, aplicar comandos de velocidade às rodas equivale a publicar no tópico `/cmd_vel` uma mensagem do tipo `geometry_msgs` que contém, como indicado em [37], uma componente para a velocidade linear e outra para a velocidade angular. Isto significa que basta aplicar as soluções do sistema dinâmico diretamente a esta mensagem.

4.3 Ambiente de Simulação

Em projetos que utilizem múltiplos robôs é comum a utilização de um simulador, para tornar fácil e prático a interpretação do comportamento dos robôs em experiências preliminares onde o objetivo é ajustar as variáveis de controlo.

Contudo o simulador implementado contém alguns pressupostos que se deve ter em consideração:

- Inexistência de erros de leitura do LRF;
- Inexistência de erros de odometria;
- Inexistência de erros de localização relativa com a simulação da câmara;
- Campo de visão da câmara do robô seguidor entre -0.457 e 0.457 radianos;
- Câmara com capacidade de detetar o marcador do robô líder até 5 metros;
- Câmara com capacidade de detetar o marcador do robô líder até um ângulo de orientação com uma diferença de ± 1.5 radianos em relação à orientação da câmara;
- Existe a possibilidade de ignorar as restrições do sistema de localização proporcionados pela câmara.

Foi utilizado o simulador *Stage* [42], suportado pelo ambiente ROS, o que torna fácil o transporte dos nós criados em ambiente de simulação para ambiente real.

4.4 Notação Usada

A notação utilizada é apresentada de seguida na figura 4.3.

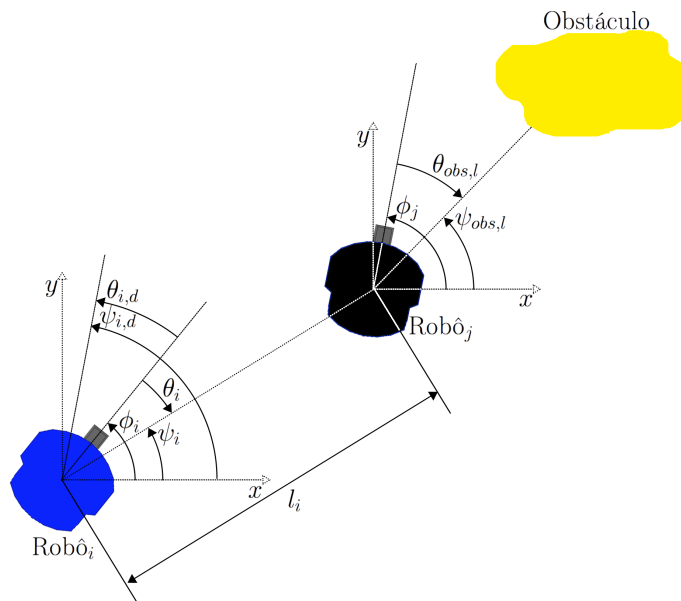


Figura 4.3: Notação usada na dissertação.

- Robô_j - Robô líder;

- Robô_{*i*} - Robô seguidor;
- l_i - Distância atual entre o robô seguidor e robô líder;
- $l_{i,d}$ - Distância desejada entre o robô seguidor e robô líder;
- ϕ_j - Orientação do robô líder;
- ϕ_i - Orientação do robô seguidor;
- $\psi_{obs,l}$ - Orientação em que o obstáculo foi detetado para a leitura l do LRF;
- $\theta_{obs,l}$ - Diferença angular entre a orientação da leitura l do LRF, $\psi_{obs,l}$, e a orientação do robô líder, ϕ_j ;
- ψ_i - Orientação em que o seguidor vê o líder;
- $\psi_{i,d}$ - Orientação desejada do seguidor, para que o seguidor permaneça em formação;
- θ_i - Diferença angular entre a orientação em que o seguidor vê o líder, ψ_i , e a orientação do robô seguidor, ϕ_i ;
- $\theta_{i,d}$ - Diferença angular desejada entre a orientação desejada do seguidor, $\psi_{i,d}$ e a orientação do robô seguidor, ϕ_i .

4.5 Aplicação das Dinâmicas de Atração ao Controlo de Formação

Nesta secção apresenta-se a aplicação das dinâmicas de atração ao controlo de robôs móveis numa formação com 2 robôs do tipo coluna e oblíqua, onde um dos robôs é o líder e outro o seguidor. Neste caso existem dois tipos de tarefas que o robô seguidor está confinado a realizar, a tarefa de evitar um obstáculo e a tarefa de convergir para a posição desejada.

4.5.1 Formação em Coluna

A formação de robôs com uma forma geométrica do tipo coluna é tipicamente utilizada no encaminhamento de um grupo de robôs para um destino definido pelo robô líder da formação.

Este tipo de formação é conseguida quando um robô seguidor se encontra atrás do robô líder, logo $\theta_{i,d} = 0$, e se encontra com uma distância l_i igual à distância desejada entre os robôs $l_{i,d}$, como apresentado na figura 4.4.

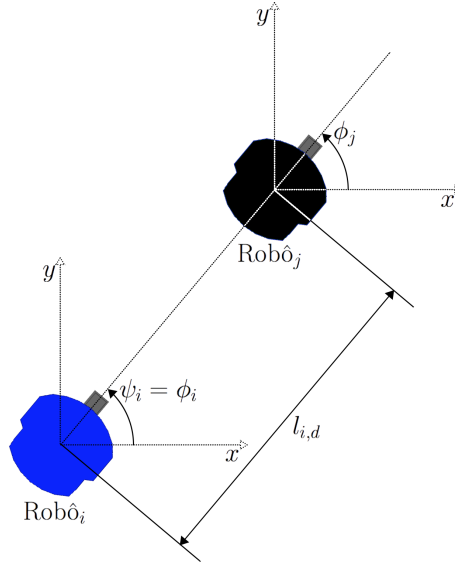


Figura 4.4: Exemplo de formação em coluna.

A equação 4.4, descreve o sistema dinâmico para a orientação do robô seguidor de forma a que seja gerada uma navegação em coluna.

$$\dot{\phi}_i = f_{coluna,i} = -\lambda_{coluna} \sin(-\theta_i) \quad (4.4)$$

O que significa que a orientação do robô é atraída para o local onde se encontra o robô líder, isto é para θ_i . Quando o robô estiver direcionado para o local onde o líder se encontra $\theta_i = 0$, o sistema dinâmico $f_{coluna,i}$ terá um valor igual a zero.

O valor de λ_{coluna} determina a força de atração máxima possível para $f_{coluna,i}$, que também corresponde à taxa de relaxamento do mesmo.

A resposta de um sistema dinâmico deste tipo é apresentada na figura 3.6, na qual se utilizou $\phi_i - \psi_i = -\theta_i$, permitindo desta forma representar o sistema dinâmico em função da orientação ϕ_i .

Uma vez determinada a contribuição angular para a navegação em coluna, para assegurar a estabilização e a permanência dos robôs em formação é necessário calcular a velocidade linear desejada para o robô seguidor, de modo a garantir que a distância entre o líder e o seguidor l_i tenda a convergir para a distância desejada $l_{i,d}$.

Pela equação 4.5 determina-se a velocidade linear desejada, onde v_j é a velocidade linear atual do líder e a variável τ_t tem como objetivo suavizar o comportamento do robô nas

acelerações.

$$v_{i,d} = \begin{cases} v_j - \frac{l_{i,d} - l_i}{\tau_i} & \text{se } l_i \geq l_{i,d} \\ -v_j - \frac{l_{i,d} - l_i}{\tau_i} & \text{caso contrário} \end{cases} \quad (4.5)$$

De modo a que a velocidade linear do robô v_i , convirja para a velocidade desejada $v_{i,d}$ faz-se uso do sistema dinâmico linear, apresentado na equação 4.6, que tem uma taxa de relaxamento determinada por α_i .

$$\dot{v}_i = g(v_i) = -\alpha_i(v_i - v_{i,d}) \quad (4.6)$$

4.5.2 Formação Oblíqua

Um grupo de dois robôs desloca-se com uma forma geométrica oblíqua, como apresentado na figura 4.5, quando o robô seguidor mantém uma diferença angular θ_i constante e igual à diferença angular desejada $\theta_{i,d} \neq 0$.

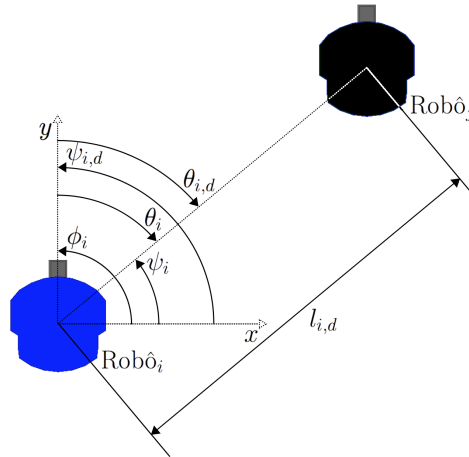


Figura 4.5: Exemplo de formação em oblíquo.

O sistema dinâmico para o controlo da orientação do robô seguidor que garante a convergência para a diferença angular desejada $\theta_{i,d}$, é dado pela equação 4.7.

$$\dot{\phi}_i = f_{oblíquo} = f_{aproximar} + f_{divergir} \quad (4.7)$$

O sistema dinâmico contém a contribuição simultânea de uma força para aproximar o robô seguidor do líder $f_{aproximar}$ e de uma força para afastar o robô seguidor do robô líder $f_{divergir}$.

A contribuição $f_{aproximar}$ é dada pela equação 4.8.

$$f_{aproximar} = -\lambda_{obliquo}\lambda_{aproximar}(l_i) \sin(\theta_{aproximar}) \quad (4.8)$$

Que constitui uma atração da variável ambiental ϕ_i para a diferença angular $\theta_{aproximar}$, determinada através da equação 4.9.

$$\theta_{aproximar} = \theta_{i,d} - \theta_i + \text{sign}(-\theta_{i,d})\gamma \quad (4.9)$$

Com uma força $\lambda_{aproximar}(l_i)$ que aumenta quando $l_i > l_{i,d}$, reduz quando $l_i < l_{i,d}$ e toma o valor 0.5 quando $l_i = l_{i,d}$, determinada por 4.10.

$$\lambda_{aproximar}(l_i) = \frac{1}{1 + e^{-\frac{l_i - l_{i,d}}{\mu}}} \quad (4.10)$$

A contribuição $f_{divergir}$ é calculada usando a equação 4.11.

$$f_{divergir} = -\lambda_{obliquo}\lambda_{divergir}(l_i) \sin(\theta_{divergir}) \quad (4.11)$$

Que constitui uma atração para a diferença angular $\theta_{divergir}$, diferença angular esta determinada pela equação 4.12.

$$\theta_{divergir} = \theta_{i,d} - \theta_i - \text{sign}(-\theta_{i,d})\gamma \quad (4.12)$$

Esta contribuição de divergência tem uma força $\lambda_{divergir}(l_i)$, determinada por 4.13.

$$\lambda_{divergir}(l_i) = 1 - \lambda_{aproximar}(l_i) \quad (4.13)$$

A variável γ presente nas contribuições 4.9 e 4.12, representa a separação das contribuições atrativas relativamente à diferença angular desejada $\theta_{i,d}$, o que significa que entre as duas contribuições atrativas existe um ângulo de 2γ , como se encontra visível na figura 4.6. Junto da variável γ está uma função sign que para valores positivos de $\theta_{i,d}$ retorna o valor -1 e para valores negativos de $\theta_{i,d}$ retorna 1, o que permite garantir que a $f_{aproximar}$ contribui para a aproximação do robô seguidor do seu líder e que $f_{divergir}$ contribui para a divergência do robô seguidor da posição do seu líder.

Na figura 4.7 encontra-se a resposta típica do sistema dinâmico para 3 situações distintas.

No primeiro caso, figura 4.7a, tem-se $l_i > l_{i,d}$, o que faz aumentar $\lambda_{aproximar}$ e por conseguinte $\lambda_{divergir}$ reduz, de tal forma que a orientação desejada para o robô seguidor, $\psi_{i,d}$, é maior que a solução constante para o sistema dinâmico naquele momento.

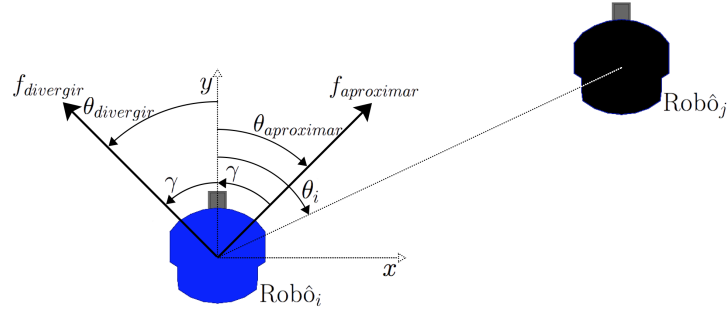


Figura 4.6: Demonstração das contribuições existentes para a dinâmica da orientação do robô em formações do tipo oblíquo.

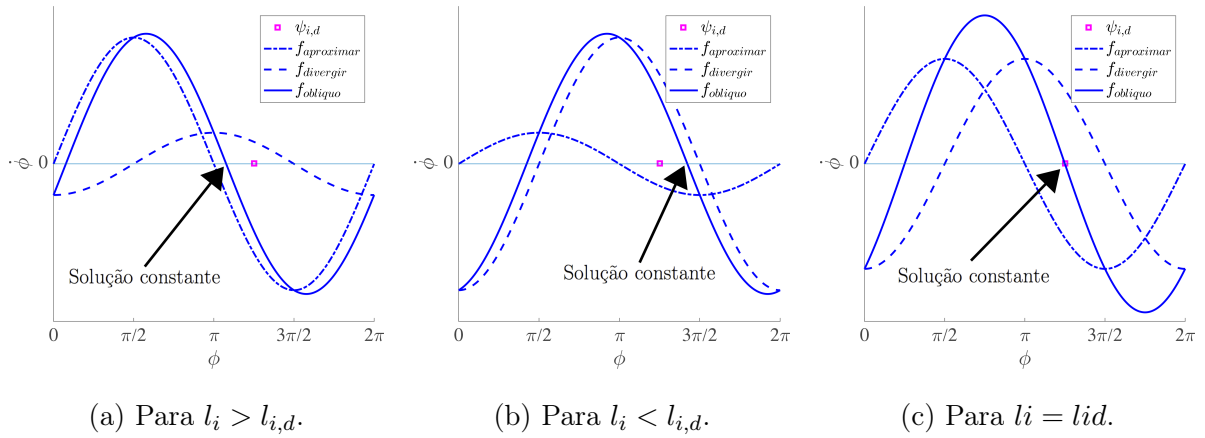


Figura 4.7: Diferentes tipos de contribuição para o controle de formação do tipo oblíquo.

Para a situação inversa, figura 4.7b, tem-se $l_i < l_{i,d}$, o que faz reduzir $\lambda_{aproximar}$ e por conseguinte aumenta $\lambda_{divergir}$, desta forma a orientação desejada para o robô seguidor, $\psi_{i,d}$, é menor que a solução constante para o sistema dinâmico nesse instante.

Quando $l_i = l_{i,d}$, figura 4.7c, as forças ficam divididas $\lambda_{aproximar} = \lambda_{divergir} = 0.5$, e por conseguinte a orientação desejada para o robô seguidor, $\psi_{i,d}$, é igual à solução constante para o sistema dinâmico.

O sistema dinâmico para a velocidade linear do robô seguidor é igual ao apresentado para a formação em coluna, equação 4.6.

4.5.3 Integração de Comportamentos

Com as dinâmicas para o controle da orientação e velocidade linear de um robô seguidor, resta englobar a capacidade do robô evitar obstáculos, como descrito na secção 3.2 do capítulo anterior.

Para a orientação do robô seguidor i , o sistema dinâmico final é dado pela equação 4.14.

$$\dot{\phi}_i = F_{obs,i} + \gamma_{coluna,i} f_{coluna,i} + \gamma_{obliquo,i} f_{obliquo,i} + f_{estoc} \quad (4.14)$$

Onde as variáveis $\gamma_{coluna,i}$ e $\gamma_{obliquo,i}$ são variáveis mutuamente exclusivas, ou seja no caso de se pretender uma formação em coluna $\gamma_{coluna,i} = 1$ e $\gamma_{obliquo,i} = 0$, e no caso de se pretender uma formação em oblíquo $\gamma_{coluna,i} = 0$ e $\gamma_{obliquo,i} = 1$. Desta forma garante-se que não existirão as duas contribuições $f_{coluna,i}$ e $f_{obliquo,i}$ em simultâneo. Como referido no capítulo anterior pode ocorrer a existência de mínimos locais, quando as variáveis $F_{obs,i}$ e $f_{coluna,i}$ ou $f_{obliquo,i}$ se anulam, daí a necessidade de incluir a função estocástica f_{estoc} para garantir que o robô seguidor não fique preso em mínimos locais.

De forma semelhante, o sistema dinâmico para a velocidade linear é dado pela equação 4.15.

$$\dot{v} = \gamma_{obs,i} g_{obs,i}(v_i) + \gamma_{coluna,i} \gamma_{obliquo,i} g(v_i) \quad (4.15)$$

Também se faz uso de variáveis mutuamente exclusivas $\gamma_{obs,i}$, $\gamma_{coluna,i}$ e $\gamma_{obliquo,i}$, para que apenas uma contribuição atribua valores a \dot{v} . Neste caso quando a direção do robô está dentro de uma região repulsiva determinada por U_{pot} , equação 3.15, obrigatoriamente a variável $\gamma_{obs,i} = 1$, $\gamma_{coluna,i} = 0$ e $\gamma_{obliquo,i} = 0$, garantindo assim que a velocidade linear é apenas determinada por $g_{obs,i}(v_i)$, equação 3.10, na existência de perigo de colisão. No caso de não existir perigo de colisão $\gamma_{coluna,i} = 1$ ou $\gamma_{obliquo,i} = 1$ e a velocidade linear é determinada por $g(v_i)$, equação 4.6.

4.6 Sumário

Este capítulo faz uma breve apresentação da nomenclatura básica utilizada pelo ROS, descreve a constituição das plataformas robóticas utilizadas, a metodologia utilizada para possibilitar a deteção de obstáculos e de marcadores visuais, o mecanismo para aplicar comandos de velocidade aos robôs, os pressupostos assumidos no desenvolvimento do simulador e apresenta também o algoritmo de controlo de formação utilizado considerando as restrições de *hardware* existentes.

Concluída a apresentação da solução para o controlo de formação, no próximo capítulo esta solução é testada com recurso a um conjunto de experiências que são analisadas.

5 Experiências e Análise e Discussão de Resultados

Com este capítulo é apresentado ao leitor as experiências realizadas em ambiente de simulação e ambiente real, onde para cada experiência pontos relevantes são mencionados.

Numa primeira fase é explicada a forma como os dados adquiridos nas experiências são tratados a fim de serem apresentados.

Posteriormente apresentam-se as experiências realizadas com 3 robôs em ambiente de simulação com o objetivo de mostrar as vantagens e limitações do sistema proposto, ao mesmo tempo que se realiza um paralelismo com as experiências em ambiente real.

Após a introdução das formas realizáveis é feita uma extensão para uma forma geométrica que recorre a 12 robôs, com este tipo de experiência pretende-se comprovar a fácil inclusão de mais robôs na formação.

O capítulo finaliza com a demonstração do comportamento do grupo de robôs sem as restrições do campo de visão limitado proporcionado pela câmara disponível.

5.1 Sistema de Cores para a Diferenciação de Robôs

Para um fácil reconhecimento dos resultados obtidos com as experiências respeitou-se o uso de um sistema de cores comum em todas as experiências realizadas. Este sistema de cores pode ser consultado na tabela 5.1.

Robô	Cor	Robô	Cor	Robô	Cor	Robô	Cor
Robô 0	●	Robô 1	●	Robô 2	●	Robô 3	●
Robô 4	●	Robô 5	●	Robô 6	●	Robô 7	●
Robô 8	●	Robô 9	●	Robô 10	●	Robô 11	●

Tabela 5.1: Identificação de robôs.

5.2 Modelo de Apresentação de Resultados

Nas experiências consideraram-se dois tipos de dados relevantes a serem apresentados, o erro de distância e o erro de orientação. A constituição destes erros apresenta-se de seguida.

5.2.1 Constituição do Erro de Distância

Considerou-se que o erro de distância é dado pela distância Euclidiana entre a posição atual e a posição desejada a que o robô seguidor vê o líder.

Pela figura 5.1 é visível que se tem disponível a diferença angular desejada e atual entre a orientação do robô seguidor e a orientação em que o seguidor vê o líder, isto é $\theta_{i,d}$ e θ_i respetivamente, e também a distância desejada entre robôs $l_{i,d}$, e a distância atual entre robôs l_i . Deste modo consegue-se determinar a distância Euclidiana que representa o erro de distância.

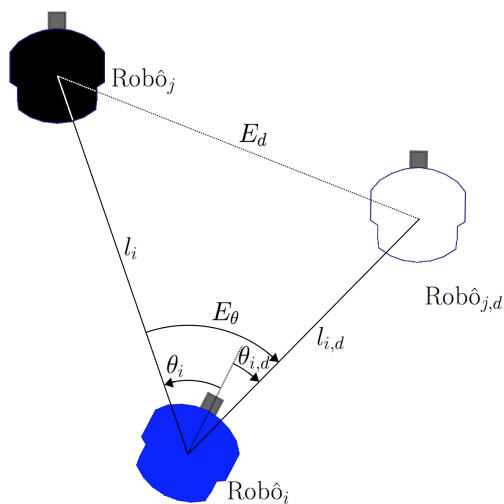


Figura 5.1: Informação disponível para determinação de erros.

O erro de distância, é dado pela equação 5.1.

$$E_d = \sqrt{l_{i,d}^2 + l_i^2 + 2l_{i,d}l_i \cos(\theta_{i,d} - \theta_i)} \quad (5.1)$$

5.2.2 Constituição do Erro de Orientação

Conhecendo a diferença angular atual e a diferença angular desejada entre a orientação do robô seguidor e a orientação em que o seguidor vê o líder, o erro de orientação é dado pela

diferença entre estas duas variáveis, erro este representado pela equação 5.2.

$$E_\theta = \theta_{i,d} - \theta_i \quad (5.2)$$

Como se faz uso de um sistema de localização descentralizado, e uma vez que não existe qualquer comunicação entre robôs, surgiu a necessidade de incluir nas amostras recolhidas, o instante temporal em que essa recolha ocorreu, passando a ser possível sincronizar as amostras de diferentes robôs num mesmo gráfico, no qual o espaço temporal é partilhado.

5.3 Experiências com 3 Robôs

As experiências para demonstrar o funcionamento do sistema implementado em ambiente de simulação e real fazem uso de 3 robôs, onde o robô líder da formação é telecomandado. Em ambos os ambientes os resultados são semelhantes, como se poderá verificar nos gráficos apresentados. Notando-se apenas uma acentuação dos erros em ambiente real provenientes de imperfeições usuais (*e.g.*, falha temporária na deteção do marcador visual, falha na estimação das velocidades). De tal forma, que apresenta-se a análise e discussão dos resultados apenas para o ambiente de simulação.

Ao longo das experiências os valores atribuídos à diferença angular desejada entre a orientação do robô seguidor e a orientação em que o seguidor vê o líder, $\theta_{i,d}$, são de -0.3, 0 e 0.3 radianos. Visto que o campo de visão da câmara está entre os -0.457 e os 0.457 radianos, o erro de orientação contém limiares que quando ultrapassados o robô seguidor deixa de detetar o seu líder.

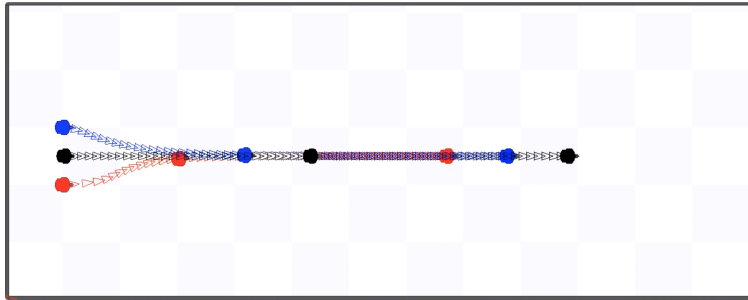
Para um $\theta_{i,d} = -0.3$ radianos significa que os limiares do erro de orientação situam-se nos -0.757 e 0.157 radianos, para um $\theta_{i,d} = 0$ radianos os limiares situam-se nos -0.457 e 0.457 radianos enquanto que para um $\theta_{i,d} = 0.3$ radianos significa que os limiares situam-se nos -0.157 e 0.757 radianos.

5.3.1 Formação em Coluna

Para demonstrar o funcionamento da formação em coluna, determinou-se que o robô 0 é o robô líder da formação, o robô 1 segue o robô 0 com uma distância relativa desejada de 1 metro ($l_{1d} = 1, \theta_{1,d} = 0$), e o robô 2 segue o robô 1 também com uma distância relativa de 1 metro ($l_{2d} = 1, \theta_{2,d} = 0$).

O conjunto de figuras 5.2 demonstram a capacidade da equipa de robôs convergir para

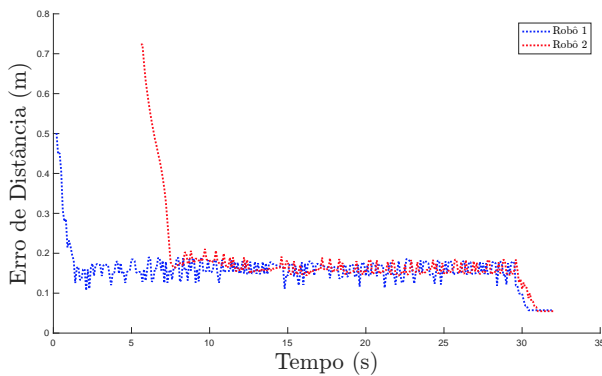
a forma geométrica do tipo coluna. É de salientar que os robôs foram posicionados de tal forma que cada robô seguidor não inclui o robô líder no seu campo de visão, e por conseguinte a forma geométrica inicial não se assemelha à forma geométrica pretendida. O percurso realizado em ambiente de simulação é apresentado na figura 5.2a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.2b.



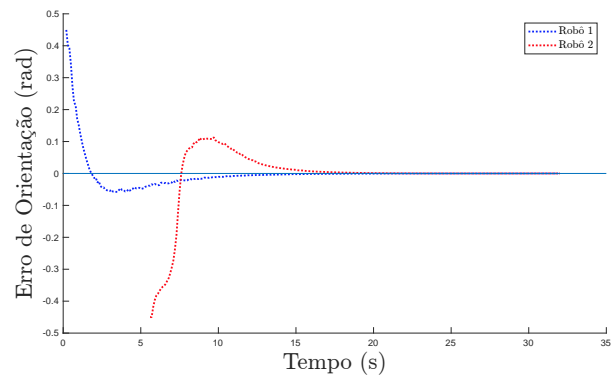
(a) Percursos realizados em ambiente simulação.



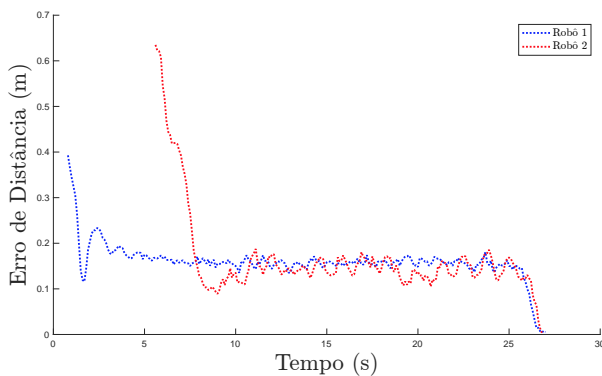
(b) Amostragem em ambiente real.



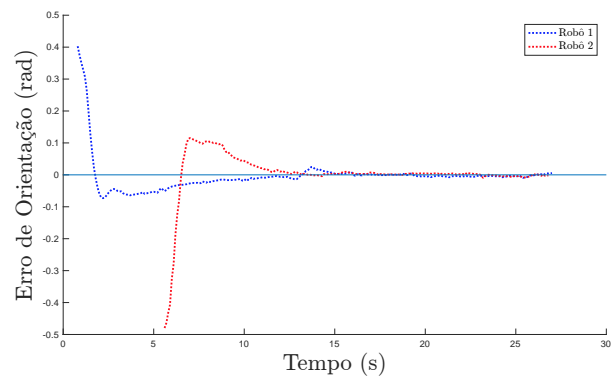
(c) Erro de distância em ambiente simulação.



(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.



(f) Erro de orientação em ambiente real.

Figura 5.2: Experiência para a forma geométrica do tipo coluna com a sala de testes sem obstáculos.

O erro de distância em simulação apresentado na figura 5.2c, permanece na casa dos 15 centímetros quando a velocidade linear do robô líder é de 0.26 m/s e a formação encontra-se estabilizada. Também se verifica que quando a velocidade linear do robô 0 reduz, o erro de distância também sofre uma redução. Aos 30 segundos o robô líder parou e verifica-se

que o erro de distância reduz até 5 centímetros. O erro não baixa mais devido às limitações mínimas de velocidade impostas.

Pela análise gráfica da figura 5.2d, percebe-se que no momento em que um robô deteta o seu líder o erro de orientação está na casa dos 0.457 radianos para o robô 1 e na casa dos -0.457 radianos para o robô 2.

Após o robô líder respetivo entrar no campo de visão do seu seguidor, para este tipo de configuração inicial, com esta forma geométrica desejada e com a velocidade linear do robô 0 de 0.26 m/s, o robô 1 consegue convergir para a posição angular pretendia em aproximadamente 12 segundos.

É de notar que o robô 2 “sente” uma forma repulsiva causada pelo robô 0 e uma força atrativa causada pelo robô 1. Por consequência aos 6 segundos verifica-se uma variação do erro de orientação, causada pelo facto de ter sido acionada a função estocástica nesse instante, pois o robô 1 entra num mínimo local. Graças à função estocástica o robô 2 consegue convergir para a posição desejada.

Com a experiência apresentada nas figuras 5.3, procura-se demonstrar a capacidade da equipa de robôs realizar rotas entre as quais o contorno de um obstáculo, como a rota apresentada na figura 5.3a que se assemelha à rota realizada em ambiente real, como apresentado na amostragem da figura 5.3b. .

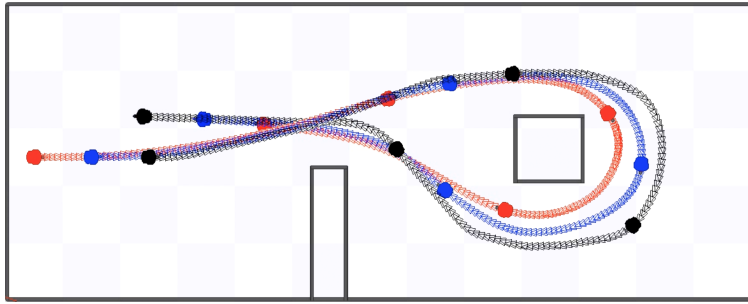
Ambos os robôs iniciam a sua movimentação com a posição desejada, mas com o intuito de evitar obstáculos, optou-se por reduzir a velocidade linear para 0.19 m/s, o que resulta numa melhoria significativa do erro de distância na ausência de obstáculos. Tal afirmação pode ser comprovada com a análise das figuras 5.2c e 5.3c.

Durante a tarefa de evitar o obstáculo quadrado, o erro de distância diminui, como se pode verificar na figura 5.2c, devendo-se isto ao facto do robô líder não aumentar a distância entre robôs ao contornar o obstáculo mas sim reduzir a distância entre eles.

Pela figura 5.3d, verifica-se que ao contornar o obstáculo quadrado o erro de orientação torna-se maioritariamente positivo pois os valores de θ_i , para $i = 1, 2$, encontram-se negativos nessa tarefa e o desejado é $\theta_{i,d} = 0$, para $i = 1, 2$.

Concluída a tarefa de evitar o obstáculo quadrado quando os robôs 1 e 2 realizam a tarefa de evitar o retângulo os valores atribuídos a θ_i , para $i = 1, 2$, são positivos e por conseguinte o erro de orientação toma valores negativos. Por fim o erro de orientação aproxima-se de zero, à medida que o grupo de robôs avança no espaço sem obstáculos.

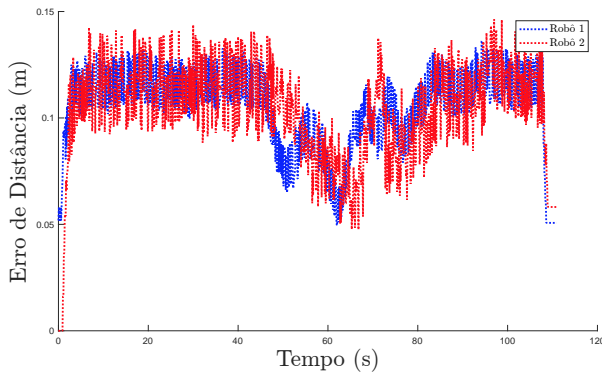
Na experiência apresentada na figura 5.4 os robôs começam novamente com a posição desejada e o objetivo é mostrar a robustez do sistema a evitar ambientes obstruídos, como



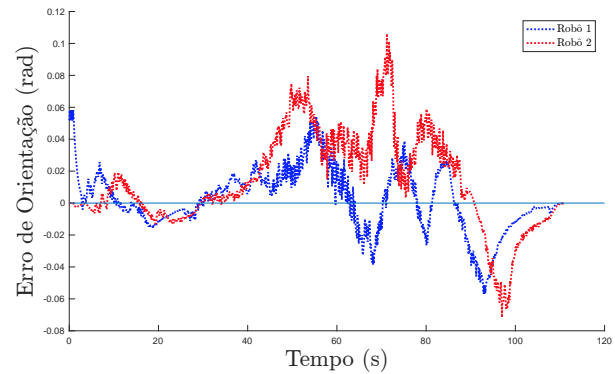
(a) Percursos realizados em ambiente simulado.



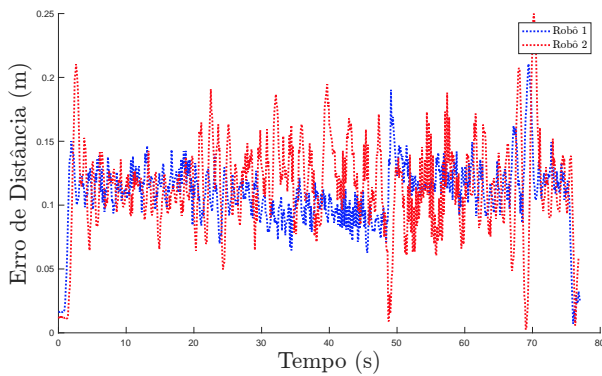
(b) Amostragem em ambiente real.



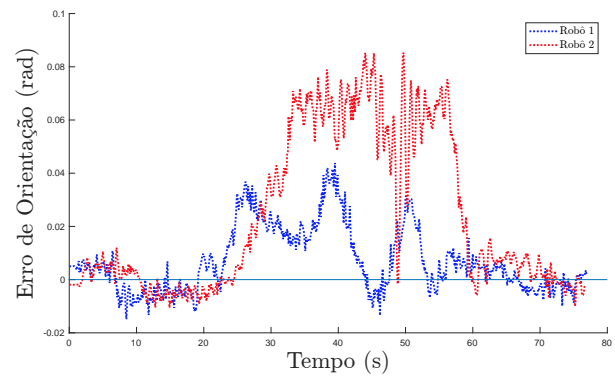
(c) Erro de distância em ambiente simulado.



(d) Erro de orientação em ambiente simulado.



(e) Erro de distância em ambiente real.



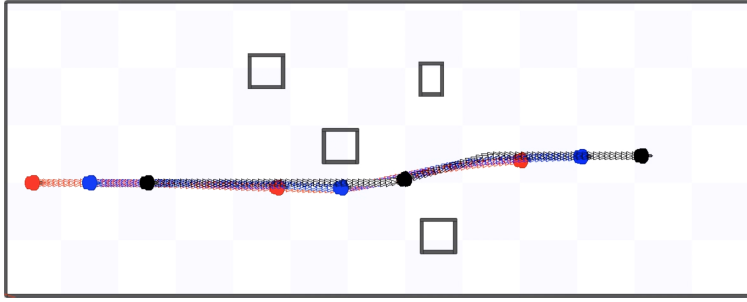
(f) Erro de orientação em ambiente real.

Figura 5.3: Experiência para a forma geométrica do tipo coluna com a finalidade de contornar um obstáculo.

apresentado na figura 5.4a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.4b.

Para uma formação em coluna, este tipo de obstáculos não proporciona uma grande perturbação. Daqui entende-se o porquê de ser vantajosa a utilização de formações do tipo coluna onde o principal objetivo seja o encaminhamento de um grupo até um determinado destino sem ocorrer perdas de robôs.

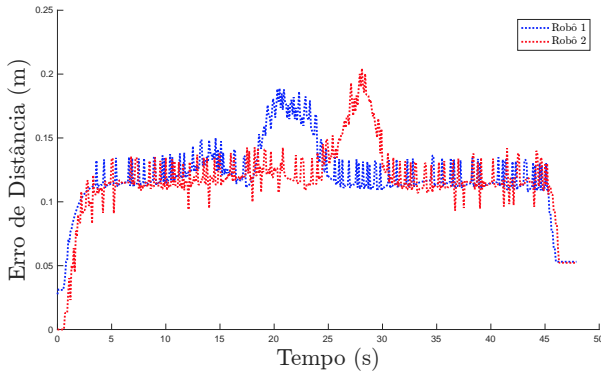
Contudo nota-se um ligeiro aumento dos erros de distância e orientação, figuras 5.4c e 5.4d, durante a existência de obstáculos próximos da rota do robô.



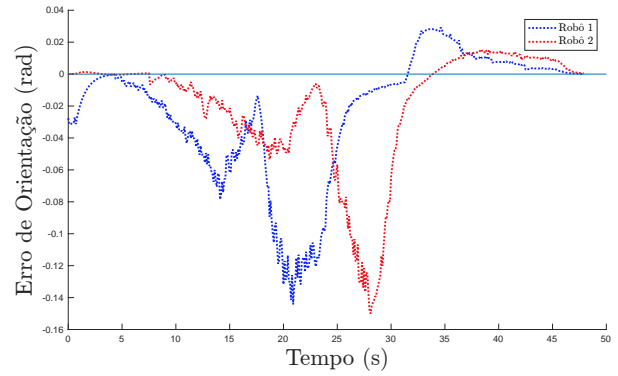
(a) Percursos realizados em ambiente simulação.



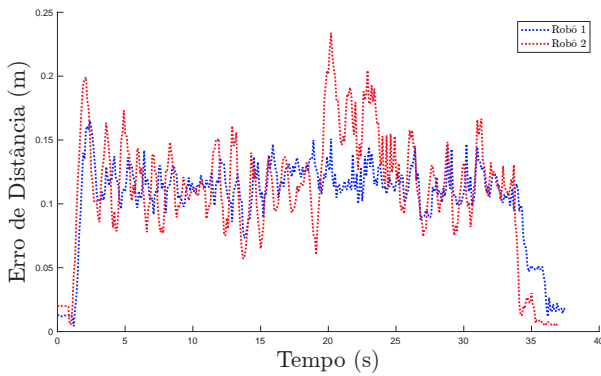
(b) Amostragem em ambiente real.



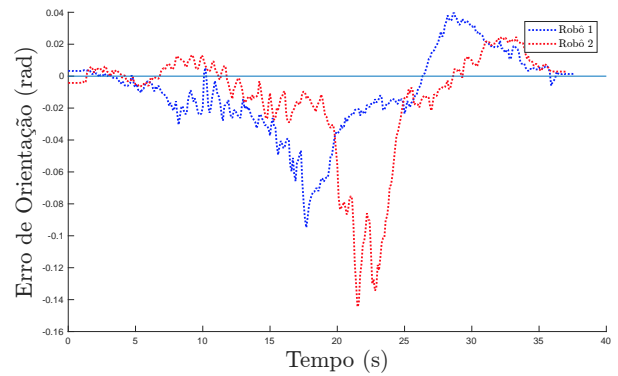
(c) Erro de distância em ambiente simulação.



(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.



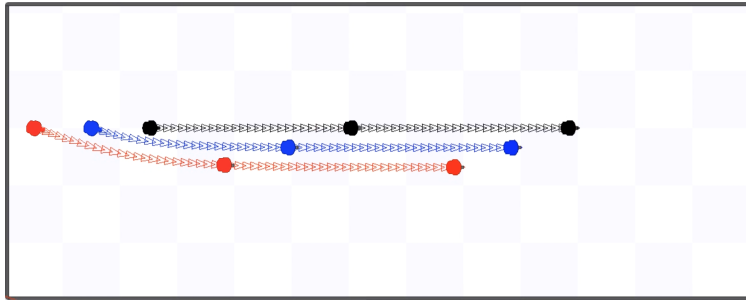
(f) Erro de orientação em ambiente real.

Figura 5.4: Experiência para a forma geométrica do tipo coluna com ambiente obstruído.

5.3.2 Formação Oblíqua

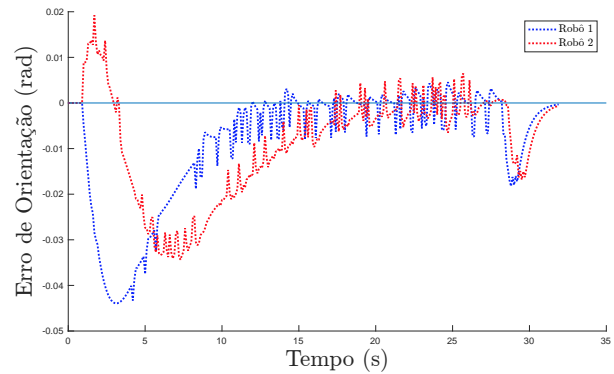
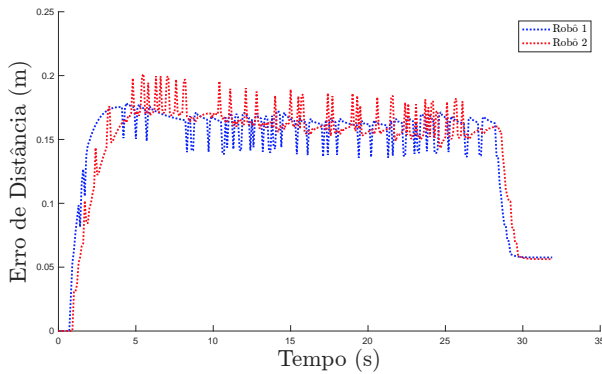
No conjunto de testes realizados para uma formação oblíqua, considerou-se que o robô 0 é o líder da formação, o robô 1 segue o robô 0 e o robô 2 segue o robô 1, ambos com uma distância relativa desejada de 1 metro e uma orientação relativa desejada de 0.3 radianos, ($l_{i,d} = 1$, $\theta_{i,d} = 0.3$ para $i = 1, 2$).

A primeira experiência, figuras 5.5, mostra a capacidade da equipa de robôs estabelecer a formação desejada começando com uma forma geométrica diferente, como apresentado na figura 5.5a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.5b.



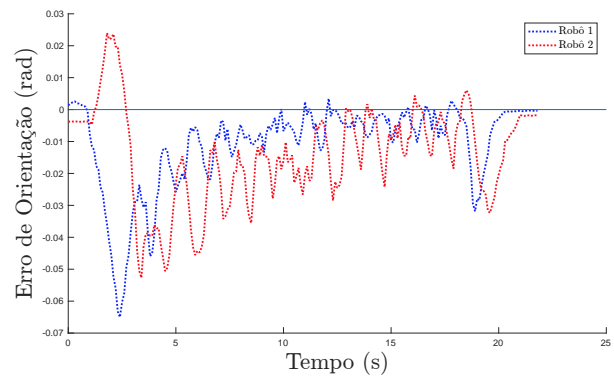
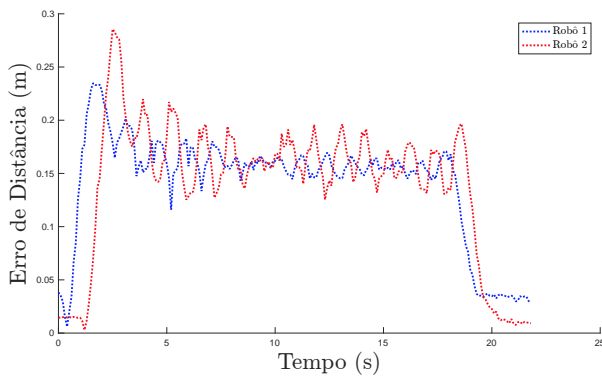
(a) Percursos realizados em ambiente simulação.

(b) Amostragem em ambiente real.



(c) Erro de distância em ambiente simulação.

(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.

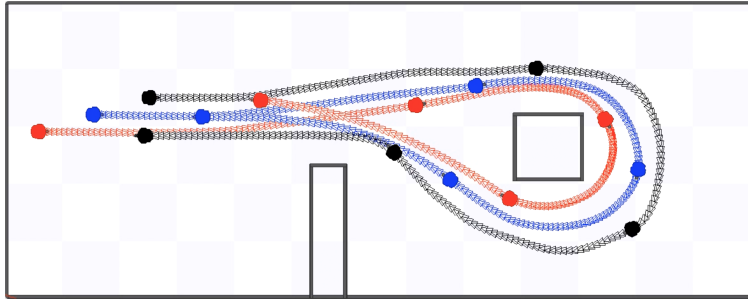
(f) Erro de orientação em ambiente real.

Figura 5.5: Experiência para a forma geométrica do tipo oblíquo com a sala de testes sem obstáculos.

Para este tipo de configuração inicial, após o robô 0 começar a deslocar-se com uma velocidade linear de 0.26 m/s , existe um aumento significativo dos erros de distância e de orientação, figuras 5.5c e 5.5d. Os erros de orientação do robô 1 estabilizam para valores desejáveis após cerca de 13 segundos do robô 0 iniciar o deslocamento. No caso do robô 2, a estabilização demora um pouco mais a estar completa uma vez que é somada à sua resposta o atraso da resposta do robô 1.

Quando o robô 0 para, o erro de distância reduz até cerca de 5 centímetros, mas com esta rápida redução o erro de orientação aumenta ligeiramente. Contudo o erro de orientação é rapidamente reduzido a valores perto de zero.

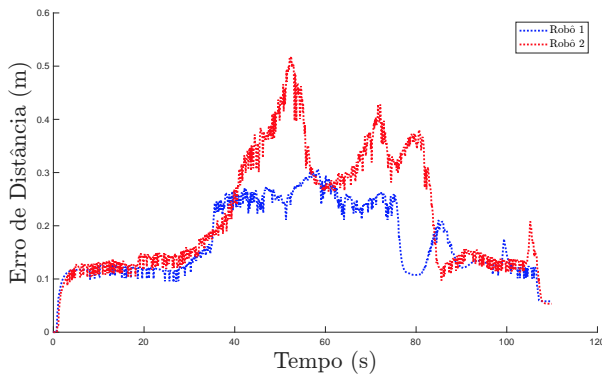
Com a experiência apresentada nas figuras 5.6, o principal objetivo é de mostrar a capacidade do sistema contornar obstáculos, como está presente no percurso da figura 5.6a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.6b.



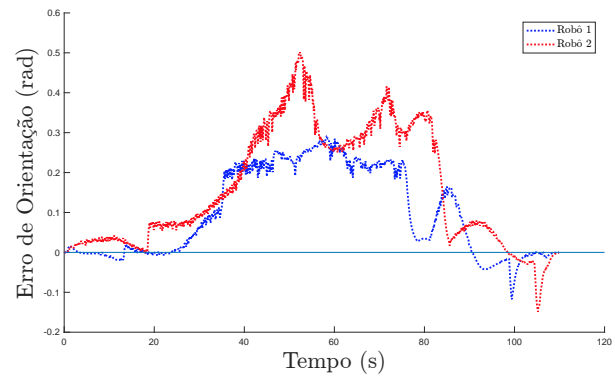
(a) Percurso realizados em ambiente simulação.



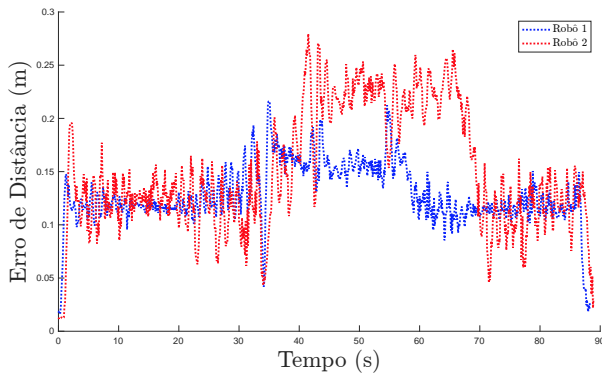
(b) Amostragem em ambiente real.



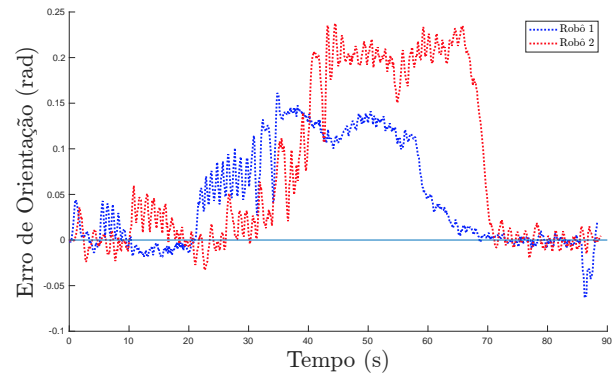
(c) Erro de distância em ambiente simulação.



(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.



(f) Erro de orientação em ambiente real.

Figura 5.6: Experiência para a forma geométrica do tipo oblíquo com a finalidade de contornar um obstáculo.

No simulador o erro de orientação, figura 5.6d, chega a alcançar os 0.5 radianos quando se encontra a contornar o obstáculo quadrado, um valor bem superior aos 0.11 radianos alcançados com a forma geométrica do tipo coluna, figura 5.3d. Como $\theta_{i,d}$ para os robôs seguidores é de 0.3 radianos o erro de orientação, tal como indicado anteriormente, pode estar entre os -0.157 e 0.757 radianos.

Com o aumento do erro angular o erro de distância também sofre um aumento, figura 5.6c.

Os robôs situados no exterior da formação, neste caso o robô 2, são os robôs que muitas vezes sentem mais perturbações pois a sua rota pode vir a diferenciar-se mais da que o robô líder da formação optou. Tal se pode verificar nos erros de distância e orientação obtidos. Nesta experiência o robô 2 é o robô que apresentou valores mais elevados para os erros.

Num ambiente mais obstruído os robôs podem decidir tomar caminhos diferentes dos que o robô líder tomou, isto para evitar colisão com obstáculos ou até mesmo entre robôs. De facto, isto é o maior constrangimento da aplicação elaborada pois em qualquer momento a escolha de tomar um caminho diferente pode refletir-se na perda do respetivo robô líder de vista.

Nas figuras 5.7, apresenta-se a passagem da equipa de 3 robôs entre um meio obstruído, realizando a trajetória presente na figura 5.7a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.7b.

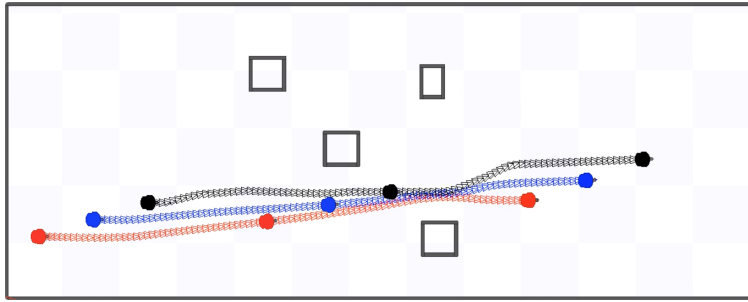
No ambiente de simulação os robôs começam novamente a movimentar-se já com a forma geométrica pretendida. Quando a equipa de robôs passa entre os obstáculos geram-se forças repulsivas, mais sentidas no robô 2 visto que a trajetória desejada coincide com a posição do obstáculo. Como pode ser verificado nos erros de orientação e de distância apresentados nas figuras 5.7c e 5.7d, respetivamente.

5.3.3 Restrições do Campo de Visão

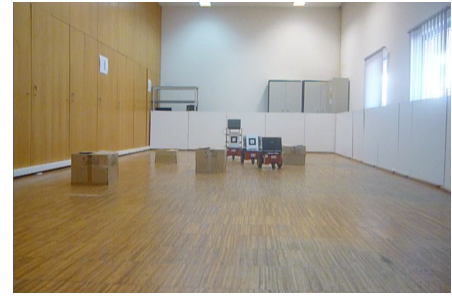
O principal problema com esta abordagem é a utilização de uma câmara com um campo de visão reduzido, o que limita o tipo de formações realizáveis. Por exemplo, a formação em linha referida por Monteiro [27] não é possível de concretizar e situações como uma variação brusca da orientação do líder podem causar a perda da localização do líder.

Com a finalidade de mostrar uma situação deste tipo de falhas, nas figuras 5.8 apresenta-se um exemplo. Nesta experiência fez-se uso de dois robôs, robôs 1 e 2, a seguirem o mesmo líder, o robô 0, realizando uma forma geométrica do tipo V, com uma diferença angular desejada de 0.3 radianos para o robô 1 e de -0.3 radianos para o robô 2, ambos com uma distância relativa desejada de 2 metros ($l_{1,d} = 2$, $\theta_{1,d} = 0.3$, $l_{2,d} = 2$, $\theta_{2,d} = -0.3$).

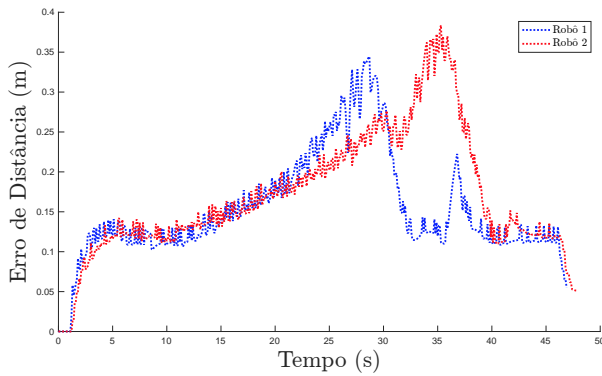
O trajeto adotado pelos robôs no simulador está presente na figura 5.8a, percurso este semelhante ao realizado em ambiente real, como apresentado na amostragem da figura 5.8b. Verifica-se que o robô 0 efetuou uma variação brusca de direção no fim de contornar o



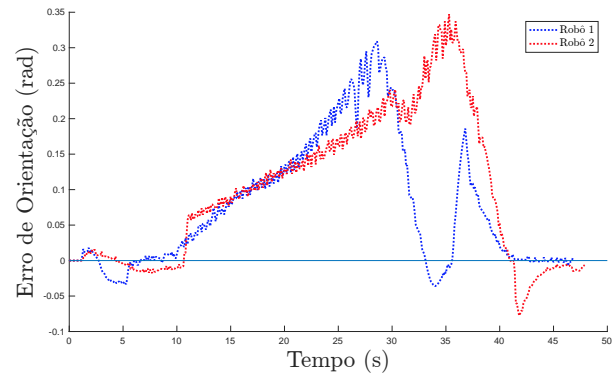
(a) Percursos realizados em ambiente simulação.



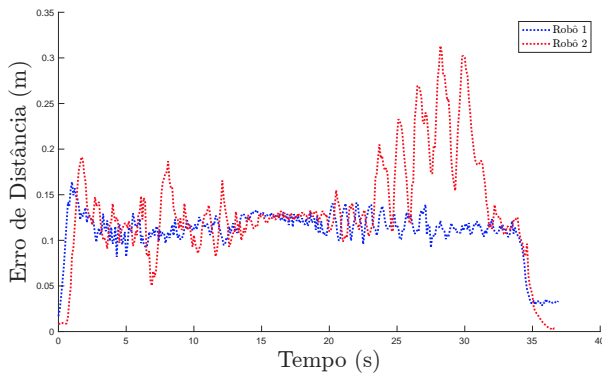
(b) Amostragem em ambiente real.



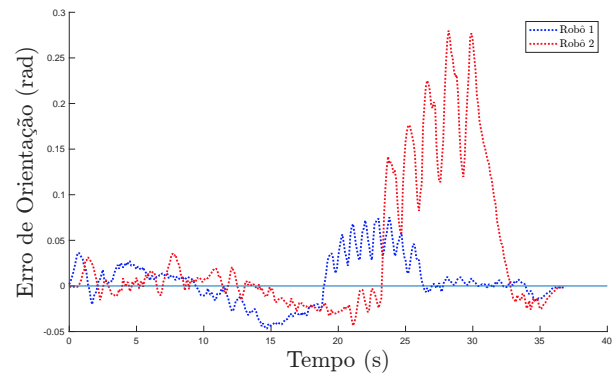
(c) Erro de distância em ambiente simulação.



(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.

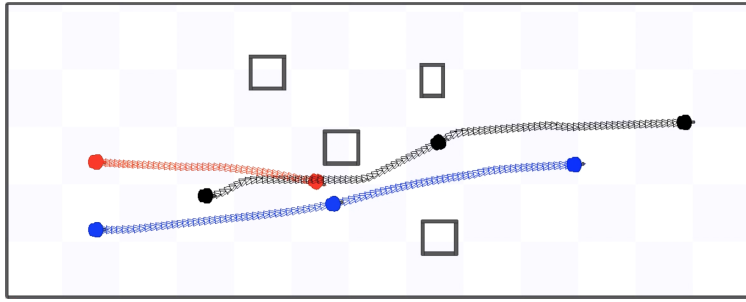


(f) Erro de orientação em ambiente real.

Figura 5.7: Experiência para a forma geométrica do tipo oblíquo com ambiente obstruído.

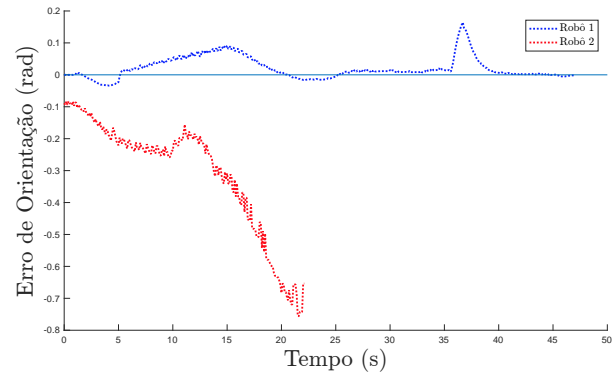
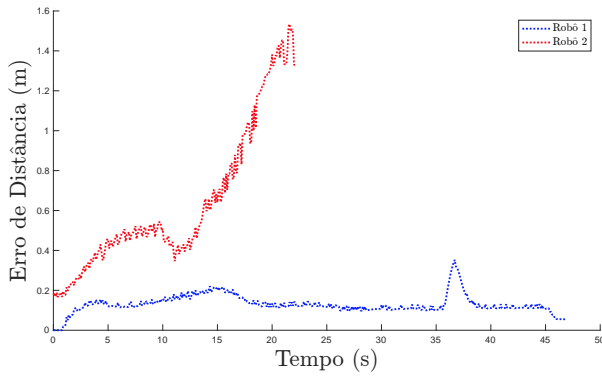
obstáculo central, variação esta de tal grandeza que o robô 0 saiu do campo de visão do robô 2. Quando ocorre a variação brusca por parte do robô 0 o robô 2 não tem capacidade de compensar o erro de orientação, figura 5.8d, pois existe um obstáculo na direção que possibilita a minimização do erro. À medida que o erro de orientação do robô 2 aumenta também se verifica um aumento do erro de distância, figura 5.8c.

A trajetória do robô 1 não foi muito influenciada pelos obstáculos o que permitiu que o robô 1 não perdesse de vista o robô 0, de tal forma que os erros convergem para valores perto de zero.



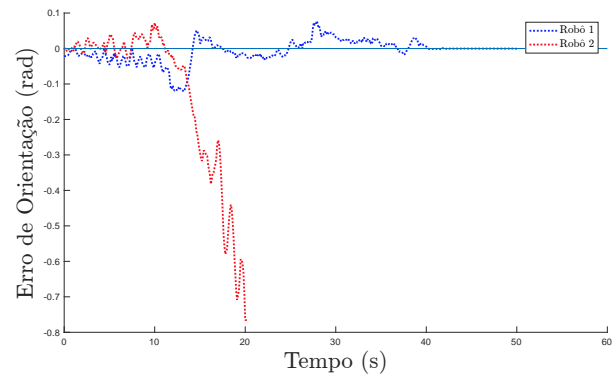
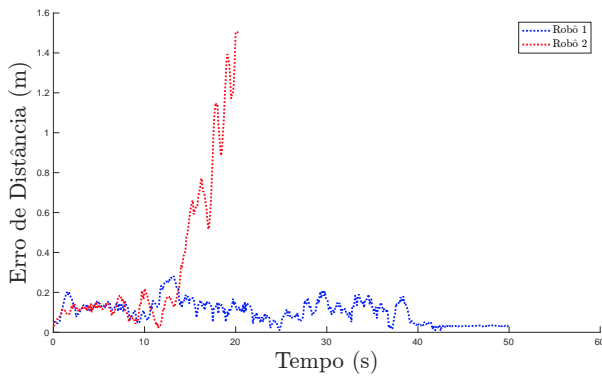
(a) Percursos realizados em ambiente simulação.

(b) Amostragem em ambiente real.



(c) Erro de distância em ambiente simulação.

(d) Erro de orientação em ambiente simulação.



(e) Erro de distância em ambiente real.

(f) Erro de orientação em ambiente real.

Figura 5.8: Experiência para a forma geométrica do tipo V com ambiente obstruído.

5.4 Experiências com 12 Robôs em Ambiente de Simulação

As experiências com 12 robôs têm como objetivo mostrar a fácil inclusão de mais robôs na equipa, pois basta indicar para cada robô que se pretende inserir na equipa o seu líder e a posição relativa pretendida.

A formação desejada para estas experiências é apresentada na figura 5.9, onde o robô 0 é o robô líder da formação e é telecomandado, o robô 1 segue o robô 0 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = 0.4$ rad, o robô 2 segue o robô 1 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = 0.4$ rad, o robô 3 segue

o robô 0 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = -0.4$ rad, o robô 4 segue o robô 3 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = -0.4$ rad, o robô 5 segue o robô 0 com $l_{i,d} = 2$ m e $\theta_{i,d} = 0$ rad, o robô 6 segue o robô 5 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = 0$ rad, o robô 7 segue o robô 6 com $l_{i,d} = 2$ m e $\theta_{i,d} = 0$ rad, o robô 8 segue o robô 4 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = -0.4$ rad, o robô 9 segue o robô 6 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = -0.4$ rad, o robô 10 segue o robô 6 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = 0.4$ rad e o robô 11 segue o robô 2 com $l_{i,d} = 1.5$ m e $\theta_{i,d} = 0.4$ rad.

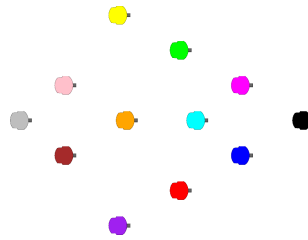


Figura 5.9: Formação desejada para experiências com 12 robôs.

Foram realizados dois tipos de experiências, uma que considera as restrições do campo de visão e outra que ignora as restrições do campo de visão.

5.4.1 Com Restrições de Campo de Visão

Com esta experiência, apresentada nas figuras 5.10, 5.11 e 5.12, pretende-se mostrar a fiabilidade do sistema proposto sem desconsiderar o campo de visão, mostrando também as limitações originadas pelo campo de visão limitado. Na figura 5.10 encontra-se o percurso realizado, com as amostragens numeradas.

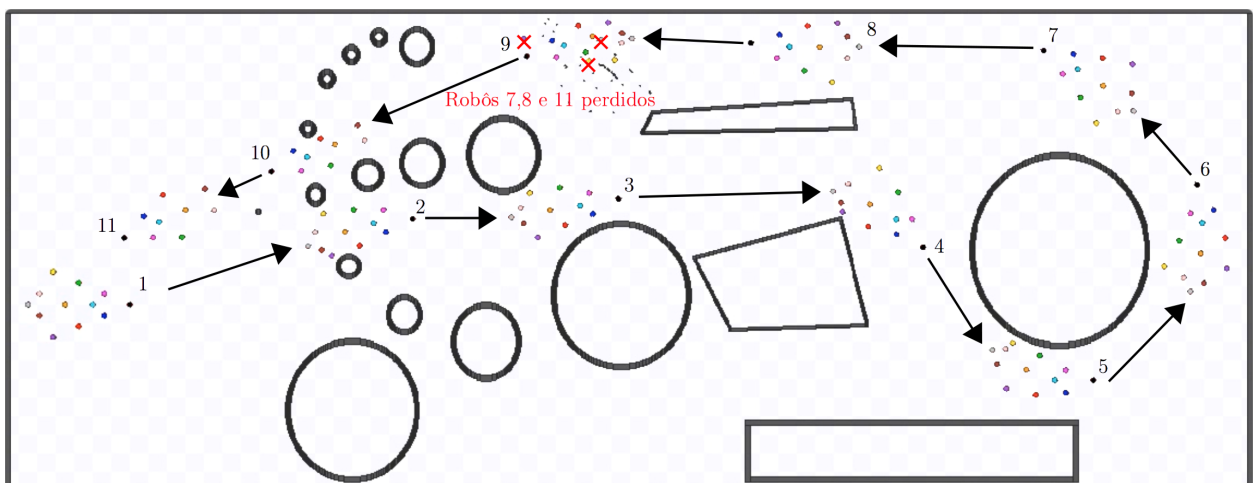


Figura 5.10: Percursos realizados na experiência com 12 robôs com as restrições do campo de visão.

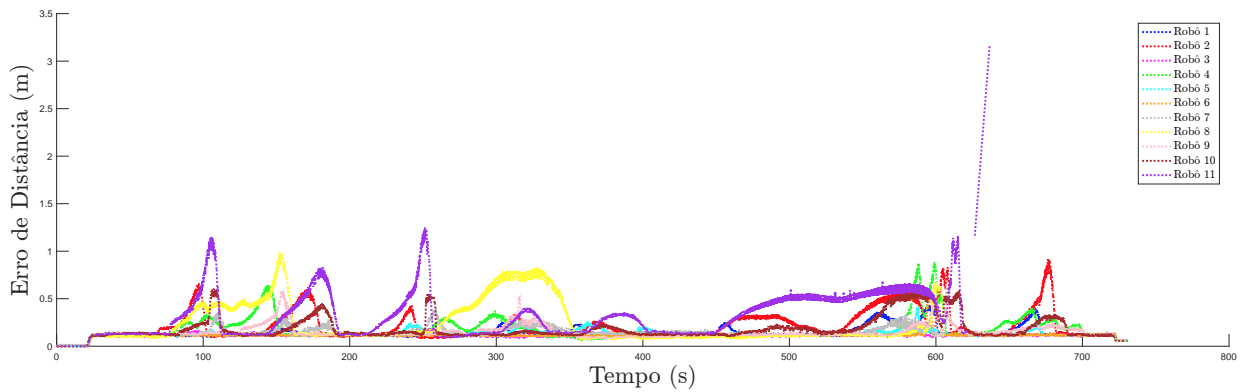


Figura 5.11: Erro de distância na experiência com 12 robôs com as restrições do campo de visão.

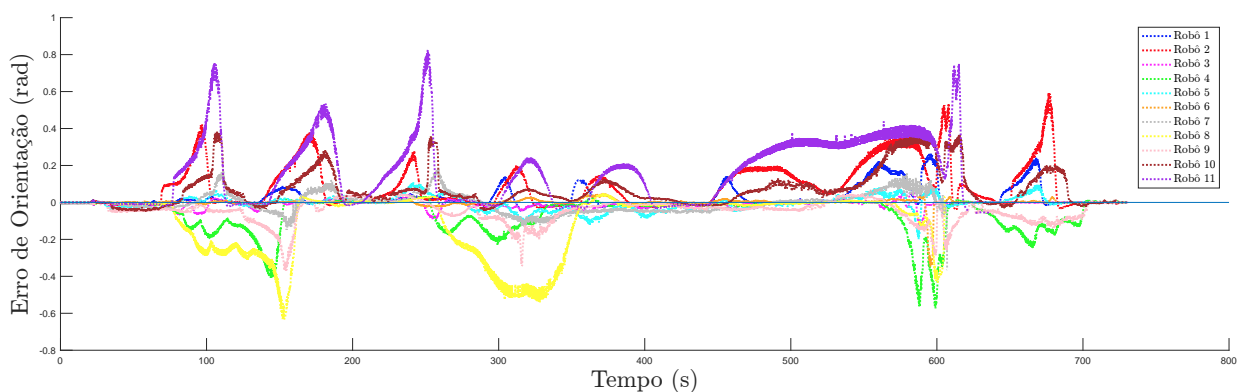


Figura 5.12: Erro de orientação na experiência com 12 robôs com as restrições do campo de visão.

Dada a configuração desejada, apresentada na figura 5.9, os valores possíveis para as diferenças angulares desejadas entre a orientação do robô seguidor e a orientação desejada do seguidor são de -0.4 , 0 e 0.4 radianos, o que significa que para a diferença angular desejada de -0.4 radianos o erro de orientação pode situar-se entre -0.857 e 0.057 radianos, para uma diferença angular desejada de 0 radianos o erro de orientação pode encontrar-se entre -0.457 e 0.457 radianos, e para uma diferença angular desejada de 0.4 radianos pode existir um erro de orientação entre -0.057 e 0.857 radianos. Para erros de orientação fora destas gamas podemos concluir que o robô seguidor perderá o seu líder.

Na primeira amostragem, os robôs estão com as posições iniciais e com a forma geométrica desejada, mantendo o líder da formação uma velocidade linear de 0.19 m/s ao longo de todo o percurso. De forma generalizada verifica-se que nas amostragens em que se encontram obstáculos os robôs exteriores têm um erro de distância, figura 5.11, e um erro de orientação, figura 5.12, maior do que nos robôs situados no interior da formação.

Perante um meio muito obstruído, amostragem 9, parte das variáveis ambientais ϕ_i to-

maior valores que colocam os respectivos robôs líderes fora do campo de visão dos seguidores, o que proporciona a ocorrência de robôs perdidos, como os robôs 7, 8 e 11, o que implica que não alcancem o destino final desejado.

5.4.2 Sem Restrições de Campo de Visão

Com o objetivo de mostrar o comportamento da equipa de robôs sem existir as restrições do campo de visão impostas pelo uso da câmara de um portátil, realizou-se a experiência apresentada nas figuras 5.13, 5.14 e 5.15. Na figura 5.13 encontra-se o percurso realizado, com as amostragens numeradas.

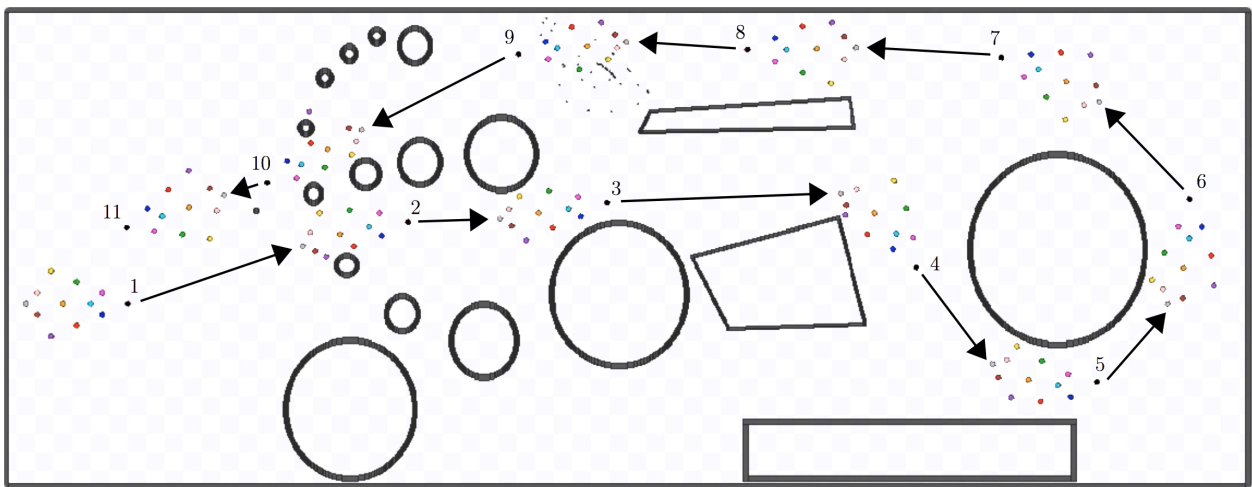


Figura 5.13: Percursos realizados na experiência com 12 robôs sem as restrições do campo de visão.

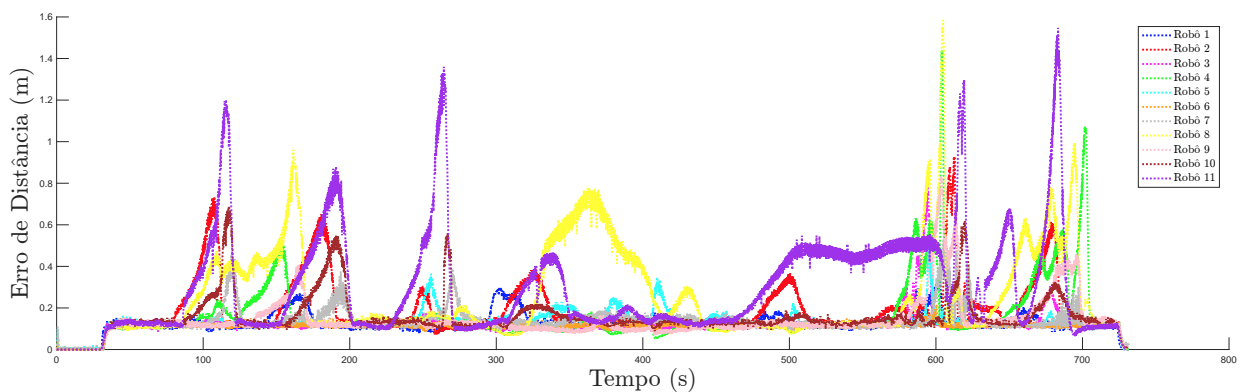


Figura 5.14: Erro de distância na experiência com 12 robôs sem as restrições do campo de visão.

Com esta experiência os erros de orientação, figura 5.15, em diversos pontos do percurso realizado pela equipa de robôs, e em particular para os robôs mais externos, foram para além

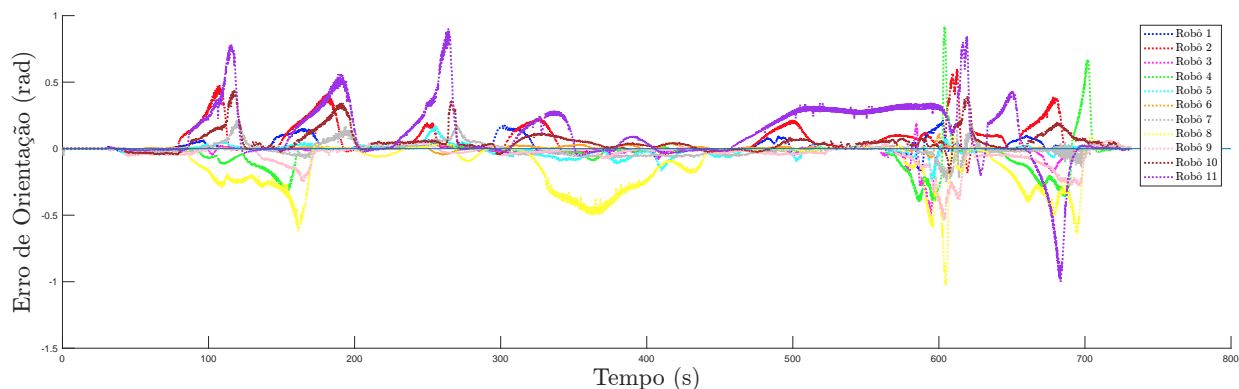


Figura 5.15: Erro de orientação na experiência com 12 robôs sem as restrições do campo de visão.

dos limites impostos na existência de uma câmara como na experiência anterior, subsecção 5.4.1.

Verifica-se também que na amostragem 9 e 10, vários robôs seguidores optaram por uma trajetória diferente da que o líder escolheu, e por conseguinte ocorreram obstáculos entre os robôs líderes e seguidores. Porém os robôs que sentiram esta perturbação acabaram por convergir para a posição desejada e por consequência a equipa de robôs também convergiu para a forma geométrica desejada como está presente na amostragem 11.

5.5 Sumário

Neste capítulo, encontra-se demonstrado o funcionamento do algoritmo implementado recorrendo a um conjunto de experiências, divididas em experiências com 3 e 12 robôs. As primeiras são concretizadas em ambientes de simulação e real. Já as experiências com 12 robôs são apenas simuladas, onde se tem como objetivos demonstrar a capacidade do algoritmo para manter formações de larga dimensão, como também demonstrar o funcionamento do algoritmo na ausência de restrições do campo de visão.

6 Conclusão e Trabalho Futuro

6.1 Conclusão

Com a abordagem implementada para o controlo de formação baseada em comportamentos, torna-se possível garantir que um grupo de robôs navegue com uma forma geométrica, definida *a priori*, em função da posição do robô líder da formação.

Dentro da estratégia de controlo implementada utiliza-se uma técnica em particular, baseada em dinâmicas de atração, onde são criados sistemas dinâmicos com o intuito de controlar as variáveis ambientais, orientação e velocidade linear da plataforma robótica.

Para que as plataformas robóticas sejam distinguidas umas das outras utilizou-se um marcador identificativo em cada uma, que com recurso a visão por computador os restantes robôs podem detetar e estimar a sua posição e respetiva velocidade. Apesar de existir um forte constrangimento causado pelo campo de visão da câmara monocular utilizada, os resultados obtidos foram satisfatórios, de tal forma que o marcador só sai do campo de visão quando ocorre uma forte variação angular de uma das plataformas.

Como se trata de uma aplicação distribuída cada robô faz uso apenas dos seus recursos computacionais, e uma vez que se trata de um sistema de localização local, permite a inexistência de um mecanismo de comunicação entre robôs, não proporcionando desta forma atrasos causados pela transmissão de dados naturalmente existentes em mecanismos de comunicação, evitando também a ocorrência de falhas de comunicação.

Foi projetado o algoritmo num simulador para possibilitar a realização de experiência em situações, por vezes até não suportáveis, que por exemplo exigem um maior número de robôs envolvidos. Para além do facto de ser muito útil para realizar ajustes de variáveis que é o caso da abordagem implementada.

Após a comprovação do funcionamento do algoritmo em ambiente de simulação, fez-se a passagem para cenários semelhantes em ambiente real com a finalidade de validar o algoritmo com as perturbações existentes neste ambiente.

6.2 Trabalho Futuro

Um dos principais constrangimentos da solução proposta está no sistema de localização que é conseguido com recurso a uma câmara monocular limitada a um campo de visão com pouca abrangência. Apesar de se obter resultados não satisfatórios apenas quando ocorre uma variação brusca da orientação dos robôs, é importante utilizar um sistema de localização que não contenha este tipo de restrições por exemplo com a inclusão de um sistema de deteção de padrões utilizando uma câmara omnidirecional, semelhante ao sugerido por Menegatti *et al.* [26] ou por Hundelshausen, Behnke e Rojas [22].

Para que o sistema seja robusto à falha de robôs, seria interessante que os robôs auto criassem agrupamentos entre si ou se desagregassem quando o robô que estão a seguir ficasse perdido. Contudo seria necessário a integração de uma arquitetura de comunicação, embora a utilização só fosse necessária para a agregação ou desagregação de robôs de agrupamentos. Esta solução pode ser conseguida com a incorporação de uma estratégia baseada na socialização entre robôs, como apresentado por Naffin e Sukhatme [30].

7 Bibliografia

- [1] Omron Adept. Research Robots - Pioneer P3-DX. <http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx>, 2016. [Último acceso: 25/08/2016].
- [2] Ashwin. How to compute intrinsic camera matrix for a camera. <https://codeyarns.com/2015/09/08/how-to-compute-intrinsic-camera-matrix-for-a-camera/>, 2015. [Último acceso: 25/08/2016].
- [3] Tucker Balch and Ronald C Arkin. Behavior-based formation control for multirobot teams. *Robotics and Automation, IEEE Transactions on*, 14(6):926–939, 1998.
- [4] Derek J Bennet and Colin R McInnes. Distributed control of multi-robot systems using bifurcating potential fields. *Robotics and Autonomous Systems*, 58(3):256–264, 2010.
- [5] Estela Bicho. *Dynamic Approach to Behavior-Based Robotics: design, specification, analysis, simulation and implementation*. PhD thesis, University of Minho, 2000.
- [6] James Bowman and Patrick Mihelich. camera_calibration. http://wiki.ros.org/camera_calibration, 2014. [Último acceso: 25/08/2016].
- [7] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. "O'Reilly Media, Inc.", 2008.
- [8] RM Kuppan Chetty, M Singaperumal, and T Nagarajan. Distributed formation planning and navigation framework for wheeled mobile robots. *Applied Sci*, 11:1501–1509, 2011.
- [9] Raul Dali Cruz-Morales, Martin Velasco-Villa, Rafael Castro-Linares, and Elvia R Palacios-Hernandez. Leader-follower formation for nonholonomic mobile robots: Discrete-time approach. 2016.

- [10] Jaydev P Desai, Jim Ostrowski, and Vijay Kumar. Controlling formations of multiple mobile robots. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 4, pages 2864–2869. IEEE, 1998.
- [11] Ivan Dryanovsk, Bill Morris, and Gautier Dumonteil. Augmented Reality Marker Pose Estimation using ARToolkit. http://wiki.ros.org/ar_pose, 2016. [Último acceso: 25/08/2016].
- [12] J Alexander Fax and Richard M Murray. Information flow and cooperative control of vehicle formations. *Automatic Control, IEEE Transactions on*, 49(9):1465–1476, 2004.
- [13] John T Feddema, Chris Lewis, and David A Schoenwald. Decentralized control of cooperative robotic vehicles: theory and application. *IEEE Transactions on Robotics and Automation*, 18(5):852–864, 2002.
- [14] Rafael Fierro, Aveek K Das, Vijay Kumar, and James P Ostrowski. Hybrid control of formations of robots. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 1, pages 157–162. IEEE, 2001.
- [15] Tully Foote, Eitan Marder-Eppstein, and Wim Meeussen. tf. <http://wiki.ros.org/tf>, 2015. [Último acceso: 25/08/2016].
- [16] Tully Foote, Eitan Marder-Eppstein, and Wim Meeussen. Transformer Class Reference. http://docs.ros.org/jade/api/tf/html/c++/classtf_1_1Transformer.html, 2016. [Último acceso: 26/08/2016].
- [17] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous robots*, 8(3):325–344, 2000.
- [18] Brian P. Gerkey, Jeremy Leibs, and Blaise Gassend. hokuyo_node. http://wiki.ros.org/hokuyo_node. [Último acceso: 26/08/2016].
- [19] Wang Guanghua, Li Deyi, Gan Wenyan, and Jia Peng. Study on formation control of multi-robot systems. In *Intelligent System Design and Engineering Applications (ISDEA), 2013 Third International Conference on*, pages 1335–1339. IEEE, 2013.
- [20] Audrey Guillet, Roland Lenain, Benoit Thuilot, and Philippe Martinet. Adaptable robot formation control: Adaptive and predictive formation control of autonomous vehicles. *Robotics & Automation Magazine, IEEE*, 21:28–39, 2014.

- [21] Ltd Hokuyo Automatic Co. Scanning range finder (SOKUIKI sensor) URG-04LX-UG01. https://www.hokuyo-aut.jp/02sensor/07scanner/urg_04lx_ug01.html. [Último acceso: 25/08/2016].
- [22] Felix v Hundelshausen, Sven Behnke, and Raúl Rojas. An omnidirectional vision system that finds and tracks color edges and blobs. In *Robot Soccer World Cup*, pages 374–379. Springer, 2001.
- [23] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5(1):90–98, 1986.
- [24] ARToolworks Support Library. Creating and training new ARToolKit markers. http://www.artoolworks.com/support/library/Creating_and_training_new_ARToolKit_markers, 2013. [Último acceso: 25/08/2016].
- [25] DC Maessen and Eberhard Gill. *Relative orbital element estimation and observability analysis for formation flying satellites using inter-satellite range measurements only*. American Institute of Aeronautics and Astronautics (AIAA), 2010.
- [26] Emanuele Menegatti, Francesco Nori, Enrico Pagello, Carlo Pellizzari, and Davide Spagnoli. Designing an omnidirectional vision system for a goalkeeper robot. In *Robot Soccer World Cup*, pages 81–91. Springer, 2001.
- [27] Sérgio Monteiro and Estela Bicho. Attractor dynamics approach to formation control: theory and application. *Autonomous Robots*, 29(3-4):331–355, 2010.
- [28] Sérgio Monteiro. *Attractor dynamics approach to formation control*. PhD thesis, University of Minho, 2007.
- [29] Eduardo Montijano, Eric Cristofalo, Dingjiang Zhou, Mac Schwager, and Carlos Sagüés. Vision-based distributed formation control without an external positioning system. *IEEE Transactions on Robotics*, 32(2):339–351, 2016.
- [30] David J Naffin and Gaurav S Sukhatme. Negotiated formations. In *Proceedings of the 8th conference on intelligent autonomous systems*, volume 1, pages 181–190, 2004.
- [31] Oladapo T Ogunbodede and Henri Huijberts. Control of unmanned aerial vehicle formations using virtual structural approach and trajectory linearisation control method. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, page 0954410015625668, 2016.

- [32] Reza Olfati-Saber and Richard M Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 3, pages 2965–2971. IEEE, 2002.
- [33] Plamen Petrov. A mathematical model for control of an autonomous vehicle convoy. *Lateral (steering)*, 5:8, 2008.
- [34] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.
- [35] Wei Ren and Randal W. Beard. Virtual structure based spacecraft formation control with formation feedback. *AIAA Guidance, Navigation, and Control Conference*, 2002.
- [36] ROS. LaserScan Message. http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html. [Último acceso: 25/08/2016].
- [37] ROS. Twist Message. http://docs.ros.org/api/geometry_msgs/html/msg/Twist.html. [Último acceso: 26/08/2016].
- [38] ROS. Writing a Simple Subscriber for Odometry. http://wiki.ros.org/evarobot_odometry/Tutorials/indigo/Writing%20a%20Simple%20Subscriber%20for%20Odometry, 2015. [Último acceso: 25/08/2016].
- [39] Lorenzo Sabattini, Cristian Secchi, and Cesare Fantuzzi. Potential based control strategy for arbitrary shape formations of mobile robots. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 3762–3767. IEEE, 2009.
- [40] Kar-Han Tan and M Anthony Lewis. Virtual structures for high-precision cooperative mobile robotic control. In *Intelligent Robots and Systems' 96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 1, pages 132–139. IEEE, 1996.
- [41] Diana D Tsankova and Nayden Isapov. Potential field based formation control in trajectory tracking and obstacle avoidance tasks. In *2012 6th IEEE International Conference Intelligent Systems*, pages 76–81. IEEE, 2012.

- [42] Richard Vaughan, Brian Gerkey, Reed Hedges, Andrew Howard, Toby Collett, Pooya Karimian, Jeremy Asher, and Alex Couture-Beil. The Stage Robot Simulator. <http://playerstage.sourceforge.net/doc/stage-svn/>. [Último acceso: 02/09/2016].
- [43] Wikipedia. Collective animal behavior. https://en.wikipedia.org/wiki/Collective_animal_behavior. [Último acceso: 04/09/2016].
- [44] Wikipedia. List of military tactics. https://en.wikipedia.org/wiki/List_of_military_tactics. [Último acceso: 04/09/2016].
- [45] Wikipedia. Camera resectioning. https://en.wikipedia.org/wiki/Camera_resectioning, 2016. [Último acceso: 25/08/2016].
- [46] Huizhen Yang and Fumin Zhang. Geometric formation control for autonomous underwater vehicles. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4288–4293. IEEE, 2010.
- [47] Jingyi Yao, Raúl Ordóñez, and Veysel Gazi. Swarm tracking using artificial potentials and sliding mode control. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):749–754, 2007.