

Marcos Felipe Santos Rabelo

Controle de Formações Homogêneas e Heterogêneas para Robôs Móveis

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica, na linha de pesquisa Robótica, Controle e Automação.

Universidade Federal do Espírito Santo – UFES
Centro Tecnológico
Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Prof. Dr. Mário Sarcinelli Filho
Coorientador: Prof. Dr. Alexandre Santos Brandão

Vitória, Espírito Santo
2018

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

R114c Rabelo, Marcos Felipe Santos, 1988-
Controle de formações homogêneas e heterogêneas para robôs móveis / Marcos Felipe Santos Rabelo. - 2018.
78 f. : il.

Orientador: Mário Sarcinelli-Filho.
Coorientador: Alexandre Santos Brandão.
Dissertação (Mestrado em Engenharia Elétrica) -
Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Robótica. 2. Robôs móveis. 3. Navegação de robôs. 4. Controle de formação. I. Sarcinelli-Filho, Mário. II. Brandão, Alexandre Santos. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

Agradecimentos

Considero esta parte a mais importante. É o momento de mostrar gratidão aos que nos dão suporte de algum modo, seja trabalhando diretamente na solução dos problemas ou apenas estando ali, compartilhando os momentos bons e principalmente os difíceis.

Agradeço meus dois orientadores, “professor” Sarcinelli e Timótiu, acima de tudo pela paciência que tiveram comigo. Não poderia ter tido melhores mentores. Saibam que vocês têm minha admiração.

Timótiu, obrigado por ter me encorajado a encarar mais um desafio. Grande parte da minha decisão foi devido a confiança que você passa. Bastava dizer “Faz isso, depois isso, aquilo e confia que vai dar certo.”, que dava certo mesmo! Mais uma vez, sem sua ajuda e suporte o caminho teria sido muito mais difícil. Obrigado por ter me aceitado novamente como orientado, e pela dose de otimista de brinde que vou incorporar daqui para frente.

Grande professor Sarcinelli, foi um prazer conhecê-lo. Obrigado pela confiança depositada em mim, e por toda ajuda durante esse tempo! Dentre suas diversas respostas, uma das mais eficientes era: “Se não está funcionando, é simples. Faça funcionar, professor!”. E depois disso funcionava mesmo. Preciso agradecer sua compreensão de minhas limitações e a sabedoria de cobrar nos momentos e medidas certas. Pode não saber mas isto foi de grande ajuda.

Agradeço o Instituto Federal Fluminense campus Itaperuna, por todo suporte e a liberação para que eu pudesse focar no curso. Foi crucial. Especialmente aos colegas da Eletrotécnica, por todo incentivo.

Colegas de turma, corpo docente e demais funcionários da UFES, que pude conviver por um tempo, obrigado pela ajuda, ensinamentos e serviços prestados. Especialmente Marisa, pelos momentos de estudo/descontração/desespero/desespero extremo que fizeram parte desta jornada.

Sou grato também à UFV, que mais uma vez me acolheu. Agradeço a todas as pessoas que pude reencontrar e conhecer em Viçosa, a família do 11, Emily e especialmente Marina por sempre estar presente. Aos caros Rodolpho, Igor e Lu, da ELT06, caminantes de longa data por essa UFV. Foi muito bom revê-los, mesmo que por pouco tempo! Os momentos de descontração/filosofias/conselhos/cafés e tudo mais foram extremamente importantes, acima de tudo para a saúde mental.

Obrigado colegas do NERO, Chakal, Valente, Kevin, Berger, Bob, João, foi muito bom trabalhar com vocês! Obrigado também membros do BDP, Sacola, Jennifer, Igor, Celso e Matheus do LOL, vocês me mostraram o significado de coragem e perseverança. Apesar de tudo foi divertido!

Agradeço os colaboradores do DEL e LabEng, destacando Amanco (A.K.A. Robson) e Eliene, pela ajuda, ideias, conversas e cafezinhos, muito obrigado por tudo!

Não poderia faltar o agradecimento aos hóspedes do Moema's Hostel: Gabriel, Andre, João, Sérgio, Rafael, Victor, Kifle e Johan. Foram muitos momentos sem dúvida memoráveis. Moema e Maria, obrigado pelos cuidados durante minha estadia em Viçosa.

Agradeço os órgãos de fomento à pesquisa, CNPQ, FAPES e FAPEMIG, que possibilitaram a realização deste trabalho.

E por fim minha família, meus pais, irmãos, obrigado por tudo. Especialmente Cecilia, Henrique, Arthur e Alice, pelo abrigo, paciência, e compreensão, vocês foram essenciais no momento mais intenso do mestrado.

E todos que não citei, afinal são muitos nomes, serei sempre grato.

Marcos Felipe Santos Rabelo

*“Quanto mais aumenta nosso conhecimento,
mais evidente fica nossa ignorância.”
(John F. Kennedy)*

Resumo

Sistemas multi-robôs possuem certas vantagens em comparação ao trabalho com um único robô. Por exemplo, permitem reduzir o tempo de execução das tarefas assim como aumentar a robustez, através de redundância. Nesse contexto, esta Dissertação de Mestrado propõe uma estrutura multicamadas para o controle de dois robôs móveis, podendo a formação ser homogênea (robôs idênticos) ou heterogênea (robôs diferentes), considerando-se veículos terrestres não tripulados (VTNT) e veículos aéreos não tripulados (VANT). A estratégia adotada considera os robôs como uma única estrutura virtual, no caso representada por um segmento de reta. Foram desenvolvidos o modelo bidimensional, considerando uma formação composta apenas por robôs terrestres (homogênea), e o tridimensional, considerando também robôs aéreos (formação heterogênea), no caso, quadrimotores. Para validação do sistema foram realizados experimentos com o robô terrestre *Pioneer 3-DX* e o quadrimotor *AR.Drone 2.0* em tarefas de posicionamento e seguimento de trajetória, para os dois modelos (2D e 3D). Para a formação heterogênea, as poses dos robôs foram obtidas através do sistema de captura de movimentos *OptiTrack*, e a partir desses dados foram obtidas também as velocidades dos veículos, por derivação. Finalmente, para demonstrar a versatilidade da estrutura de controle proposta, foram realizadas simulações envolvendo uma nova formação homogênea, agora com dois quadrimotores.

Abstract

Multi-robot systems have advantages over working with a single robot. It allows reducing the task-execution time and more robustness, through redundancy. In such a context, this Master Thesis proposes a multilayer control structure to control a formation of two mobile robots, considering homogeneous (identical robots) or heterogeneous (different robots) formations. Unmanned ground vehicles (UGV) and unmanned aerial vehicles (UAV) are considered. The strategy adopted considers the two robots as a single virtual structure, in this case represented by a straight line segment. The two-dimensional model was developed, considering a homogeneous formation composed just by terrestrial robots, as well as the three-dimensional one, now considering a UGV and a quadrotor (a heterogeneous formation). For system validation, experiments were carried out with the *Pioneer 3-DX* terrestrial robot and the *AR.Drone 2.0* quadrotor, in positioning and trajectory-tracking tasks, for both 2D and 3D models. For the heterogeneous formation, the poses of the two robots were obtained through the *OptiTrack* motion capture system, and from these data the velocities were also obtained, by differentiation. Finally, to demonstrate the versatility of the proposed control structure, simulations were carried out involving a new homogeneous formation, now composed by two quadrotors.

Sumário

1	INTRODUÇÃO	17
1.1	Estado da Arte	20
1.2	Objetivos	22
1.3	Estrutura do Trabalho	23
2	ROBÔS DA FORMAÇÃO	25
2.1	<i>Pioneer 3-DX</i>	25
2.2	<i>AR.Drone 2.0</i>	27
3	ESTRUTURA E CONTROLE DA FORMAÇÃO	29
3.1	Caracterização da formação	29
3.1.1	Formação 2D	29
3.1.2	Formação 3D	31
3.2	O controle multicamadas	33
3.2.1	Controlador da formação	35
3.2.2	Matriz Inversa do Jacobiano	35
3.3	Análise de estabilidade do controlador	36
4	RESULTADOS	39
4.1	Formação 2D	39
4.1.1	Metodologia	39
4.1.2	Controle de Posição	40
4.1.3	Seguimento de trajetória circular	42
4.2	Formação 3D	43
4.2.1	Metodologia	43
4.2.2	Controle de Posição	45
4.2.3	Seguimento de Trajetória	49
4.2.3.1	Lemniscata de Bernoulli	50
4.2.3.2	Elipse	51
4.2.3.3	Elipse com aterrissagem e decolagem	56
4.2.4	Simulação da formação homogênea aérea	60
5	CONCLUSÕES	67
	REFERÊNCIAS	71
	APÊNDICE A – OPTITRACK	75

Lista de ilustrações

Figura 1 – Exemplos de sistemas multirobôs.	18
Figura 2 – Robôs utilizados na pesquisa.	25
Figura 3 – Representação das variáveis do <i>Pioneer 3-DX</i>	26
Figura 4 – Representação das variáveis dos <i>ArDrone 2.0</i>	28
Figura 5 – Configuração da formação como segmento de reta bidimensional.	29
Figura 6 – Configuração da formação como segmento de reta tridimensional.	31
Figura 7 – Estrutura do sistema de controle multicamadas.	33
Figura 8 – Diagrama de blocos do sistema de controle.	34
Figura 9 – Percorso realizado para tarefa de posicionamento 2D.	41
Figura 10 – Evolução temporal dos erros das variáveis da formação.	41
Figura 11 – Evolução temporal das velocidades dos robôs.	42
Figura 12 – Percursos dos robôs e do ponto de referência da formação no seguimento da trajetória circular 2D.	43
Figura 13 – Erros associados às variáveis de formação durante o seguimento da trajetória 2D circular.	44
Figura 14 – Velocidades desejadas e efetivamente desenvolvidas pelos robôs da formação.	44
Figura 15 – Sequência de quadros do momento da aterrissagem no experimento de posicionamento.	46
Figura 16 – Vistas do experimento de posicionamento para formação heterogênea.	46
Figura 17 – Posições desejadas e realizadas dos robôs ao longo do experimento de posicionamento 3D.	47
Figura 18 – Variáveis da formação ao longo do percurso.	48
Figura 19 – Erros das variáveis de formação para tarefa de posicionamento 3D.	49
Figura 20 – Sinais de controle desejados e realizados dos robôs ao longo do experimento de posicionamento 3D.	50
Figura 21 – Trajetória Lemniscata desejada e realizada pela formação.	51
Figura 22 – Posições desejadas e realizadas dos robôs para a trajetória Lemniscata.	52
Figura 23 – Variáveis da formação ao longo da trajetória Lemniscata.	53
Figura 24 – Erros das variáveis de formação para trajetória Lemniscata.	53
Figura 25 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória Lemniscata.	54
Figura 26 – Trajetória Elíptica desejada e realizada pela formação.	55
Figura 27 – Trajetória elíptica desejada e realizada separada em duas partes.	56
Figura 28 – Posições desejadas e realizadas pelos robôs na trajetória elíptica.	57
Figura 29 – Variáveis da formação ao longo da trajetória elíptica.	58

Figura 30 – Erros das variáveis de formação para trajetória elíptica.	58
Figura 31 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória elíptica.	59
Figura 32 – Sequência de quadros respectivamente do momento da aterrissagem e decolagem no experimento de seguimento de trajetória elíptica.	60
Figura 33 – Percorso desejado e realizado no experimento de trajetória elíptica com aterrissagem e decolagem durante o percurso.	61
Figura 34 – Posições desejadas e realizadas dos robôs ao longo do percurso elíptico com aterrissagem e decolagem durante o experimento.	62
Figura 35 – Variáveis da formação ao longo do percurso.	63
Figura 36 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória.	64
Figura 37 – Erros das variáveis de formação para trajetória elíptica.	65
Figura 38 – Resultados da simulação da formação homogênea com dois VANT.	66
Figura 39 – Esquema de ambiente para experimentos com o <i>OptiTrack</i>	75
Figura 40 – Ferramentas para calibração das câmeras do <i>OptiTrack</i>	76
Figura 41 – Robôs com seus marcadores e os respectivos corpos rígidos criados no Motive 2.0.	77

Lista de tabelas

Tabela 1 – Parâmetros dos robôs para os experimentos.	39
Tabela 2 – Ganhos do controlador da formação 2D.	40
Tabela 3 – Parâmetros das variáveis de formação desejadas para controle de posicionamento 2D.	40
Tabela 4 – Parâmetros das variáveis de formação desejadas para trajetória 2D circular.	42
Tabela 5 – Ganhos do controlador da formação 3D.	45
Tabela 6 – Variáveis de formação desejada para o experimento de posicionamento 3D.	45
Tabela 7 – Parâmetros das variáveis de formação desejada para a Lemniscata.	51
Tabela 8 – Parâmetros das variáveis de formação desejada para a elipse.	54
Tabela 9 – Parâmetros das variáveis de formação desejada para o experimento de trajetória elíptica com aterrissagem.	56
Tabela 10 – Parâmetros das variáveis de formação desejada para a formação homogênea.	61

1 Introdução

Trabalhar com robôs móveis traz vantagens, entre elas a segurança. Por exemplo, é mais interessante utilizar um robô ao invés de um humano em áreas potencialmente perigosas, como ambientes radioativos. Além da segurança, há a questão da maior eficiência em certas situações, como em missões de localização, suporte e resgate de vítimas de acidentes entre outros. Isto porque com o auxílio de robôs pode-se cobrir uma maior área de busca em menor tempo, como mencionado em (SCHNEIDER; WILDERMUTH, 2017). E com o avanço das pesquisas, eles vêm ganhando cada vez mais autonomia.

No entanto, a robótica ainda abrange muitos desafios. Da montagem de um robô até a execução de determinada tarefa é preciso solucionar diferentes problemas, sejam eles mecânicos, computacionais, de comunicação ou de controle. Na robótica móvel, a movimentação do robô acrescenta navegação autônoma à lista de desafios a serem vencidos. Neste caso, espera-se que o robô se locomova em algum ambiente, podendo ser estruturado, semiestruturado ou não estruturado. Ambientes estruturados são aqueles completamente conhecidos, que permitem a navegação do robô sem a preocupação com modificações físicas no espaço. No ambiente semiestruturado o ambiente é parcialmente conhecido, ou seja, partes podem modificar-se com o tempo, como uma sala onde entram e saem pessoas. Por fim no ambiente não estruturado não há informações prévias, logo é completamente desconhecido (TZAFESTAS, 2014).

Para tais navegações, o robô normalmente necessita atingir determinada posição ou percorrer determinada trajetória. Essas tarefas são conhecidas como controle de posição e seguimento de trajetória, respectivamente. Na tarefa de posicionamento, o objetivo é alcançar uma posição desejada, sem definição de tempo, percurso ou velocidade a serem desenvolvidos. No seguimento de trajetória, por sua vez, são definidos um conjunto de pontos consecutivos que o robô deve percorrer, que formam um caminho específico. Além disso, a velocidade e o tempo do percurso são definidos. Quando não existe restrição temporal associada à velocidade, tem-se o chamado seguimento de caminho (SICILIANO; KHATIB, 2016).

Conforme explicado em (SICILIANO; KHATIB, 2016), o controle de robôs para as tarefas mencionadas é um objeto de estudo relevante, visto que a movimentação dos robôs é essencial para sua aplicação em diversas áreas. Especificamente, trabalhos envolvendo múltiplos robôs (do inglês *Multiple Robots System*) vêm sendo estudados principalmente quando se deseja:

- redução de tempo para execução das missões;
- maior robustez do sistema por redundância;

- redução de custo e simplificação do sistema com a possibilidade de utilizar vários robôs menos especializados ao invés de um muito robusto.

Como exemplo, podem ser citados transporte de carga (MASON; BÜLTHOFF; STEGAGNO, 2016) e mapeamento (LI et al., 2014). Esse campo de estudo envolvendo múltiplos agentes divide-se em dois grupos: Enxame e Formação (ou Cooperação) de robôs. O enxame é um sistema com muitos robôs de capacidade limitada, e mínima comunicação entre eles. Ele é inspirado em formações da natureza, como grupos de abelhas, formigas, pássaros e peixes. A filosofia do enxame é o alcance do objetivo global através de ações individuais independentes. Na formação, por sua vez, geralmente são grupos menores de robôs que cooperam entre si intencionalmente. Os membros compartilham informações entre si e realizam tarefas em conjunto, com uma missão em comum. A Figura 1 traz exemplos dos dois tipos.

Um dos aspectos dos sistemas multirobôs é a capacidade de seus agentes. Nesse sentido, o grupo pode ser homogêneo ou heterogêneo. A formação é homogênea quando os membros são iguais e possuem as mesmas funções, sendo vantajoso em relação à redundância, uma vez que todos os robôs podem executar as mesmas tarefas, sendo facilmente substituíveis. Se algum elemento do grupo possui características diferentes (como os tipos de sensores embarcados), a formação é heterogênea. Neste caso é comum a distribuição de tarefas considerando a compatibilidade do robô com a mesma, resultando em otimização de resultados. No âmbito construtivo, pode ser complicado criar um robô capaz de atender todas as demandas de algum projeto. Ao invés disso, uma saída é desenvolver diferentes robôs que, através da cooperação, realizarão o objetivo global superando as limitações individuais.

Um veículo terrestre não tripulado (VTNT) cooperando com um veículo aéreo não tripulado (VANT) representa uma formação heterogênea. Tal configuração pode ser

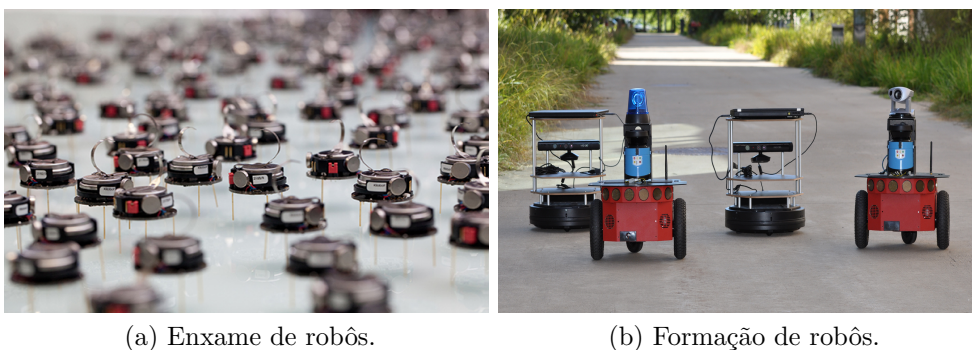


Figura 1 – Exemplos de sistemas multirobôs.

(a) Fonte: Disponível em <<https://www.sheffield.ac.uk/faculty/engineering/news/prog-robots-1.553813>>. Acesso em 20 nov. 2018.

(b) Fonte: Disponível em <<http://robot.cfp.co.ir/en/educationdetail/958>>. Acesso em 20 nov. 2018.

utilizada para identificar alvos a partir de diferentes ângulos (CAI; KOUSUKE, 2015), transporte e manipulação de objetos (NGUYEN; GARONE, 2016; ARBANAS et al., 2016), missões de vigilância (KHALEGHI et al., 2015), busca e resgate em construções danificadas (HOOD et al., 2017), detecção de radiação (CHRISTIE et al., 2017), inventário em armazéns (ZHANG et al., 2016), agricultura de precisão (TOKEKAR et al., 2016) entre outros exemplos. Isto demonstra o alto potencial de aplicações mesmo para uma formação simples VTNT-VANT, que é o objeto de pesquisa nesta dissertação.

Para que as tarefas descritas anteriormente possam ser executadas de modo autônomo, é necessário um sistema de controle. Os modelos mais comuns são o centralizado e o descentralizado. Quando uma única unidade recebe, processa e envia comandos para os membros da formação, o controle é centralizado. Este modo é mais indicado para coordenação de robôs em tarefas que exijam maior sincronismo e percepção ampla do ambiente. Porém, é vulnerável a falhas na central de comando, e na prática possui restrições quanto a comunicação à medida que o número de robôs cresce.

Por outro lado, se cada robô gera seu próprio sinal de controle independentemente, o controle então é descentralizado. Significa que não há uma única unidade responsável pelo controle da formação, tornando o sistema mais robusto em relação a falhas individuais. No entanto, nesse caso os robôs devem ser capazes de compartilhar entre si informações acerca de suas respectivas posições, o que em situações reais pode nem sempre ser possível (YAMCHI; ESFANJANI, 2017).

Considerando o controle da formação, foram desenvolvidas várias estratégias, como líder-seguidor, estrutura virtual, baseada em comportamento, campos potenciais e consenso (LIANG et al., 2016), sendo as três primeiras mais conhecidas. Na prática, é comum a combinação de dois ou mais métodos de modo a eliminar ou reduzir as desvantagens individuais.

Na formação líder-seguidor, um robô é definido como o líder, que será seguido pelos restantes, os chamados seguidores (como detalhado em (BRANDÃO et al., 2009; BOEL; MARINICA; SARLETTE, 2015; LI; DUAN; HUANG, 2009)). O líder então, é a referência da formação, e os seguidores apenas repetem as ações do líder. Para isso, o seguidor precisa conhecer as velocidades do líder. O aspecto negativo desta abordagem é que, na possível falha do líder, toda formação será prejudicada, impedindo a execução da tarefa.

Já o método da estrutura virtual não necessita de um elemento central. A formação é considerada uma única estrutura formada pelos robôs (veja (KITTS; MAS, 2009; ADAMEK; KITTS; MAS, 2015; BRANDÃO et al., 2015)), os quais se movem no sentido de preservar a forma, que pode ser rígida (não se altera) ou flexível (pode alterar seu formato).

Por fim, a estratégia baseada em comportamento consiste em estabelecer para cada robô uma ação específica, no intuito de manter a formação durante a navegação (BRANDÃO; SARCINELLI-FILHO, 2016; ROSALES et al., 2016; BALCH; ARKIN, 1998; ANTONELLI; ARRICHIELLO; CHIAVERINI, 2010). Os comportamentos são programados como módu-

los normalmente ativados pelas leituras dos sensores ou outros comportamentos. Ações como seguir parede, encontrar objeto, retornar à posição inicial e seguir alvo são exemplos de comportamentos que podem ser implementados nos robôs. A principal dificuldade desta abordagem está na escolha de uma ação específica e manutenção da coesão do grupo. Uma das soluções é estabelecer uma hierarquia na decisão do comportamento para atingir o objetivo global (SICILIANO; KHATIB, 2016).

De acordo com as características da tarefa em questão, assim como os recursos disponíveis, as diferentes metodologias abordadas devem ser analisadas para a escolha da mais apropriada. Contudo, independentemente do modelo adotado, há um problema comum a todos: o erro nas medidas de posição através da odometria (SIEGWART et al., 2011). A imprecisão dos sensores inerciais embarcados (do inglês *Inertial Measurement Unit* - IMU) que possuem preços mais acessíveis e, principalmente, a integração numérica das velocidades para gerar a posição são responsáveis por erros cumulativos que propagam-se ao longo do percurso (SANTOS, 2017). Uma das formas de mitigar esses efeitos inerentes ao sistema é o uso da visão computacional através do processamento de imagens, amplamente comentado em (AL-KAFF et al., 2017).

Entre as soluções baseadas em imagens, um exemplo mais voltado à pesquisa é o sistema de captura de movimento (Mocap - *Motion Capture*). Uma das aplicações é na modelagem e validação de modelos de robôs, como foi feito em (ZHU; ZHU; XU, 2016) e (SCHANG et al., 2018) para um VANT. O Mocap opera com câmeras posicionadas em diferentes ângulos para captura e processamento de imagens de um mesmo ambiente. Deste modo, é possível estimar com alta precisão a postura de um objeto de interesse em relação ao referencial global, permitindo assim a aquisição da postura dos robôs. Exemplos comerciais desse sistema são *OptiTrack* (<www.optitrack.com>), *Vicon Motion Systems* (<www.vicon.com>) e *Qualisys* (<www.qualisys.com>).

1.1 Estado da Arte

Os primeiros estudos sobre sistemas robóticos multiagentes remontam à década de 80, e desde então sistemas multirrobôs vêm sendo tópico de pesquisa, muito devido aos diversos desafios que proporciona (SICILIANO; KHATIB, 2016). Como citado anteriormente, formações de robôs podem ser utilizadas em diversas aplicações. Por exemplo, a cooperação entre um robô terrestre e outro aéreo foi abordada em (SURMANN; BERNINGER; WORST, 2017), para mapeamento e localização em regiões de desastre através da fusão de imagens aéreas de câmera monocular e dados de um sensor laser embarcado no veículo terrestre. A proposta foi validada através de experimentos.

Já em (LUDVIGSEN et al., 2016) é utilizada uma formação heterogênea para aumentar o alcance e capacidade da comunicação de veículos autônomos submarinos para explorações e vigilância subaquática. Por sua vez, em (CHRISTIE et al., 2017) foram feitos experimentos

com uma formação heterogênea VANT-VTNT para buscas em ambientes radioativos. O VANT faz imagens aéreas, que são processadas e permitem ao robô terrestre traçar uma rota. O veículo terrestre possui um sensor LiDAR e um medidor de radiação para coletar informações do local. A classificação dos objetos das imagens é através de segmentação semântica, com aprendizagem de máquinas, e o planejamento de caminho utiliza o algoritmo A*.

Uma estratégia de líder-seguidor para robôs *car-like* é proposta em (GUILLET et al., 2017), com algumas adaptações. Cada robô se comunica apenas com seus vizinhos imediatos, e, para aumentar a confiabilidade, um deles é considerado líder local. Além disso, para reduzir os efeitos de erro cumulativo, há dois líderes globais, um no início e outro no final da formação. Foram realizadas simulações e experimentos para validação do modelo. O foco é aplicações em agricultura, voltado para tratores. No entanto, a estrutura genérica do sistema permite seu aproveitamento em outras áreas.

Dentre as estratégias de controle de formação pesquisadas, a líder-seguidor é a mais recorrente. Em (ABAUNZA et al., 2017), um VANT em trajetória circular segue um VTNT através de controle baseado em quatérnios. De acordo com os autores, este método evita efeitos indesejados como *gimbal lock*, singularidades e descontinuidades nos ângulos que podem atrapalhar o controle. Normalmente não há comunicação entre os seguidores e o líder, sendo necessários métodos de estimativa da velocidade. Em (LIANG et al., 2018), para uma formação líder-seguidor, foi feita a estimativa da posição e velocidades do robô líder através de observadores de estado. Os dados foram obtidos por câmera embarcada no seguidor. O sistema foi validado através de simulações e experimentos, utilizando robôs não-holonômicos. Em (WANG et al., 2017) os observadores de estado foram utilizados para determinar os parâmetros intrínsecos e extrínsecos da câmera, e o controle foi baseado nas imagens obtidas. A estabilidade do sistema foi provada utilizando a teoria de Lyapunov, e o sistema foi validado através de experimentos. Na pesquisa de (OYEKAN, 2016) também é proposto um controle líder-seguidor baseado apenas em visão, porém sem utilizar sistemas de predição. O modelo é tolerante a um certo atraso ocasionado pelo próprio processo de aquisição e processamento das imagens.

No método de estruturas virtuais não há a dependência de um líder, porém a comunicação entre os agentes é um fator importante, conforme explicado na seção anterior. Em (ABBASI; MOOSAVIAN; NOVINZADEH, 2017), é proposto um controle PID descentralizado para formação de veículos aéreos de asas fixas por estruturas virtuais. O sincronismo é ajustado por lógica *fuzzy*. No intuito de garantir a navegação mesmo sem comunicação entre os agentes, as posições dos vizinhos são estimadas a partir de sensores passivos. A validação da proposta foi feita por simulações. Em (ROSALES et al., 2016), experimentos foram realizados para um controle centralizado de uma formação aérea para tarefa de posicionamento e seguimento de trajetória. O método de estruturas virtuais foi utilizado, porém com características do método baseado em comportamento. Foram

definidas subtarefas para formação, que foram hierarquizadas utilizando o espaço nulo da matriz Jacobiano que relaciona as velocidades dos robôs com as da formação. O sistema de controle foi expandido para formações heterogêneas.

Alguns pesquisadores buscam aproveitar as melhores características de cada modelo, a fim de ampliar a capacidade do sistema. Um controle de chaveamento automático entre os métodos líder-seguidor, baseado em comportamento e estruturas virtuais para navegação é proposto em (QIANG et al., 2018). Para simplificação e economia de energia do sistema, a localização dos robôs é feita por UWB (*Ultra Wide Band*), que transmite grande quantidade de dados digitais pulsados em um amplo espectro de frequência, com baixo consumo de energia para pequenas distâncias. Experimentos foram realizados para validação do sistema.

Em relação ao planejamento de caminho, o ideal é encontrar a melhor rota para a formação, o que irá resultar em economia de tempo ou energia, por exemplo. Várias pesquisas abordam este problema com técnicas de otimização. Em (WAN; LU; FAN, 2017), o método Húngaro é proposto para posicionamento dos robôs na formação em um modelo líder-seguidor semi-centralizado. No entanto, o sistema foi validado apenas em simulações. Já em (ALONSO-MORA; BAKER; RUS, 2017) é aplicada uma técnica de otimização com restrições para controle centralizado de formações aéreas ou terrestres, com validação através de experimentos. Segundo o autor, o algoritmo mantém a eficiência com o aumento de agentes na formação. Algo semelhante foi feito em (URCOLA et al., 2017), com acréscimo de fusão sensorial de dados dos agentes da formação utilizando um filtro de Kalman, para melhorar a informação de localização. Em (JASIM; GU, 2018), foi feita uma comparação entre um controlador subótimo H_∞ e um *Integral Backstepping* para VANT. Foram feitas simulações onde o primeiro demonstrou melhor desempenho aos distúrbios e erros de modelagem.

Um grande obstáculo para o controle de formação, porém, é a comunicação. Atrasos, falhas ou falta de sincronismo no *link* podem prejudicar significativamente o desempenho da formação, principalmente quando há trabalho coordenado envolvido, como transporte cooperativo de carga. Em (YAMCHI; ESFANJANI, 2017; PENG et al., 2018) é proposto um controle descentralizado para navegação de formação, considerando atraso de comunicação entre os agentes, através de controle preditivo. Em ambos os trabalhos as propostas foram validadas através de simulações.

1.2 Objetivos

O objetivo desta dissertação é desenvolver uma estrutura de controle para uma formação de dois robôs, seja ela homogênea ou heterogênea, utilizando o mesmo controlador. No caso da formação homogênea, serão dois robôs terrestres do tipo uniciclo, considerando a navegação em terreno plano (apenas duas dimensões). E a formação heterogênea será

composta de um robô terrestre também uniclo e um robô aéreo do tipo quadrimotor.

O controle da formação será baseado no método de estruturas virtuais, onde os robôs são vistos como uma forma geométrica, navegando de modo a manter esta formação. Para o caso de dois robôs, a forma escolhida é um segmento de reta bidimensional a princípio, e posteriormente tridimensional, tendo como extremidades cada robô.

Para validação do controlador proposto, serão feitas simulações e experimentos utilizando o robô terrestre *Pioneer 3-DX* da MobileRobots e o aéreo *AR.Drone 2.0* da Parrot Drones SAS. As formações executarão tarefas de posicionamento e seguimento de trajetória.

1.3 Estrutura do Trabalho

A organização desta Dissertação segue a estrutura abaixo:

Capítulo 1: Introdução

Este capítulo apresenta uma visão geral sobre robótica móvel, suas vantagens e desafios, incluindo o estado da arte referente à cooperação de robôs e os objetivos da pesquisa.

Capítulo 2: Robôs da formação

Neste capítulo são descritos os robôs utilizados para os experimentos, bem como seus modelos cinemático e dinâmico necessários para aplicação do controle.

Capítulo 3: Estrutura e controle da formação

Aqui é definida a configuração das formações como um segmento de reta, através do método de estruturas virtuais, tanto para formação homogênea terrestre como para heterogênea. Além disso, é discutida a estrutura de controle multicamadas proposta e verificada a estabilidade do controlador pela teoria de Lyapunov.

Capítulo 4: Resultados

Os resultados dos experimentos e as discussões acerca dos mesmos são mostrados neste capítulo. Nele foram analisados experimentos de controle de posicionamento e seguimento de trajetória para a formação homogênea terrestre e a heterogênea. Para demonstrar a versatilidade da plataforma, foram acrescentados resultados de uma simulação de seguimento de trajetória para formação homogênea com dois quadrimotores.

Capítulo 5: Conclusões

Este capítulo apresenta a conclusão e propostas de continuação da pesquisa em trabalhos futuros.

Após estes capítulos é introduzido um apêndice a respeito do sistema de captura de movimentos *OptiTrack*, utilizado nos experimentos com formação heterogênea.

2 Robôs da formação

Os robôs utilizados para simulações e experimentos nesta pesquisa foram o *Pioneer 3-DX*, da MobileRobots, e o *AR.Drone 2.0*, da Parrot Drones SAS, ambos apresentados na Figura 2. Neste capítulo são descritas as principais características, o modelo e o controlador de cada robô.

2.1 *Pioneer 3-DX*

O *Pioneer 3-DX* é um robô terrestre do tipo unicycle com boa confiabilidade. São duas rodas de tração independente e uma roda castor para apoio da estrutura. Esta configuração permite que ele gire em torno do próprio eixo, conferindo três graus de liberdade ao robô. Ele também possui sensores que geram informações como posição (x,y) , ângulo de guinada e velocidades linear e angular atuais (SANTANA; BRANDÃO; SARCINELLI-FILHO, 2016). Sua comunicação com um computador externo é através da porta serial RS-232, e os sinais de controle enviados ao robô são expressos pelo vetor

$$\mathbf{u}_d = [v_d \quad \omega_d],$$

onde v_d e ω_d são respectivamente, a velocidade linear e angular desejadas.

O modelo cinemático de um robô do tipo unicycle como o *Pioneer 3-DX* pode ser descrito por

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\psi) & -a \sin(\psi) \\ \sin(\psi) & a \cos(\psi) \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad (2.1)$$



(a) Pioneer 3-DX



(b) AR.Drone 2.0

Figura 2 – Robôs utilizados na pesquisa.

(a) Fonte: Disponível em <<http://neuromorphs.net/nm/wiki/2016/ppr16>>. Acesso em 13 ago. 2018.

(b) Fonte: Disponível em

<<https://www.parrot.com/global/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition>>.

Acesso em 05 jul. 2018.

onde $[\dot{x} \ \dot{y}]^T$ é o vetor de velocidades nos eixos x e y globais, ψ é o ângulo de guinada do robô, $[v \ \omega]^T$ é o vetor de velocidades linear e angular e a é a distância do ponto de interesse (ponto cuja posição é controlada) ao ponto médio da linha que une os eixos das rodas de tração, como mostrado na Figura 3.

Invertendo a relação descrita na Equação (2.1), é possível encontrar as velocidades linear e angular correspondentes às velocidades desejadas em relação aos eixos x e y global. Logo, a partir desta relação, as velocidades de referência (\dot{x}_r e \dot{y}_r) geradas pelo controle da formação (será explicado no Capítulo 3) serão convertidas nos sinais de controle de referência \mathbf{u}_r para o robô.

Entretanto, este controle é para um sistema ideal, pois desconsidera a massa do robô e os efeitos da inércia na navegação, ou seja, é como se o robô atingisse instantaneamente as velocidades desejadas, o que na prática não acontece. Uma forma de tornar este modelo cinemático mais realista é acrescentar a compensação dinâmica. O compensador calcula novos sinais de controle \mathbf{u}_d levando em conta os parâmetros dinâmicos do robô. O modelo de compensador dinâmico utilizado segue o proposto em (MARTINS; SARCINELLI-FILHO; CARELLI, 2017), cuja estabilidade foi provada pelos autores.

Em consequência, o sinal de controle realmente enviado ao robô é

$$\mathbf{u}_d = \mathbf{H}(\dot{\mathbf{u}}_r + \mathbf{K}_{d1} \tanh(\mathbf{K}_{d2} \tilde{\mathbf{u}})) + \mathbf{C}\mathbf{u}_r, \quad (2.2)$$

onde $\tilde{\mathbf{u}}$ é o erro das velocidades linear e angular, respectivamente $\tilde{v} = v_r - v$ e $\tilde{\omega} = \omega_r - \omega$, \mathbf{K}_{d1} e \mathbf{K}_{d2} são matrizes de ganho diagonais definidas positivas e \mathbf{H} e \mathbf{C} são matrizes relacionadas ao modelo dinâmico do *Pioneer 3-DX* (MARTINS; SARCINELLI-FILHO; CARELLI, 2017). Considerando os parâmetros dinâmicos $\theta_i, i = 1, \dots, 6$, as matrizes \mathbf{H} e \mathbf{C} ficam

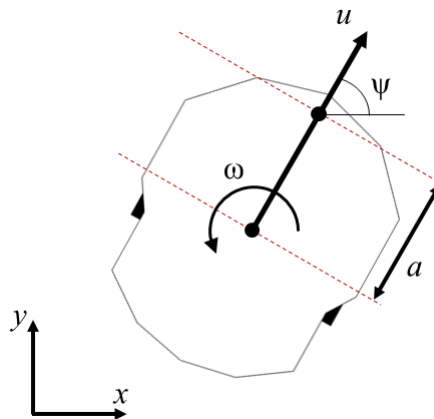


Figura 3 – Representação das variáveis do *Pioneer 3-DX*.

Fonte: Autoria própria.

$$\mathbf{H} = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix}$$

e

$$\mathbf{C} = \begin{bmatrix} \theta_4 & -\theta_3\omega \\ \theta_5\omega & \theta_6 \end{bmatrix}.$$

Os parâmetros θ são referentes às constantes físicas do robô, tais como massa, raio das rodas, resistência elétrica dos motores entre outras. Entretanto, algumas constantes não são obtidas facilmente, como por exemplo a constante eletromotriz dos motores e o coeficiente de atrito. Entretanto, existem diferentes técnicas para estimação do parâmetros. Nos experimentos realizados neste trabalho foram utilizados os mesmos θ identificados em (MARTINS, 2009) através do método dos mínimos quadrados para o *Pioneer 3-DX* sem acessórios extras (*laser* e/ou câmera).

2.2 AR.Drone 2.0

O *AR.Drone 2.0* é um quadrimotor que foi vendido como brinquedo¹, e pode ser controlado por *smartphones* ou *tablets* através de rede Wi-fi (SANTANA; BRANDAO; SARCINELLI-FILHO, 2015). Ele vem equipado de fábrica com um processador ARM Cortex A8 de 1 GHz com sistema operacional Linux 2.6.32, além de sensores embarcados que permitem a medição de variáveis nos seis graus de liberdade do robô. Há também um controlador interno responsável pela decolagem, voo pairado (*hovering*) e aterrissagem.

O quadrimotor possui seis graus de liberdade conforme mostra a Figura 4, porém apenas quatro sinais de controle, o que configura um sistema subatuado. Segundo o Guia do Desenvolvedor (*Software Development Kit - SDK*) (PISKORSKI et al., 2012), os sinais de comando enviados ao robô são

$$\mathbf{u} = [u_\phi \quad u_\theta \quad u_z \quad u_\psi]^T \in [-1, 1],$$

onde:

- u_ϕ controla o ângulo de rolagem, responsável pela movimentação esquerda-direita;
- u_θ controla o ângulo de arfagem, que resulta na movimentação para frente e para trás;
- u_z controla a velocidade vertical;

¹ Segundo informações no site do fabricante (<<https://www.parrot.com>>), o *AR.Drone 2.0* não é mais comercializado.

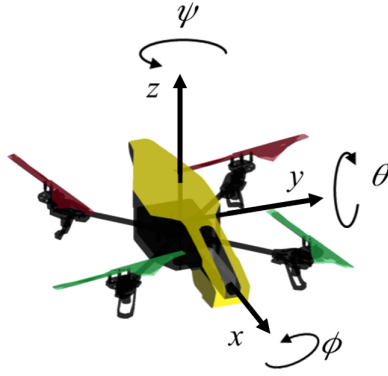


Figura 4 – Representação das variáveis dos *ArDrone 2.0*.
Fonte: Autoria própria.

- u_j é responsável pela taxa de guinada, que rotaciona o robô em relação ao eixo z .

O modelo dinâmico e controlador para o quadrimotor considerado nesta pesquisa será o mesmo desenvolvido em (BRANDÃO; SARCINELLI-FILHO; CARELLI, 2013). Considerando o vetor das variáveis de postura do robô $\mathbf{d} = [x \ y \ z \ \phi \ \theta \ \psi]^T$, o modelo dinâmico simplificado do *AR.Drone 2.0* é dado pela equação²

$$\mathbf{M}(\mathbf{d})\ddot{\mathbf{d}} + \mathbf{C}(\mathbf{d}, \dot{\mathbf{d}})\dot{\mathbf{d}} + \mathbf{G} = \mathbf{u} - \mathbf{D},$$

sendo $\mathbf{M}(\mathbf{d})$ a matriz rotacional de inércia, $\mathbf{C}(\mathbf{d}, \dot{\mathbf{d}})$ a matriz da força de Coriolis e \mathbf{D} representa os possíveis distúrbios provocados por atrito, efeitos aerodinâmicos, rajadas de vento entre outros fenômenos. \mathbf{G} é o vetor da Força da Gravidade que atua nos três eixos cartesianos e \mathbf{u} é o vetor dos sinais de controle do sistema.

O controlador é dado por

$$\mathbf{u}_d = \ddot{\mathbf{d}}_d + \mathbf{K}_1 \tanh(\mathbf{K}_2 \dot{\tilde{\mathbf{d}}}) + \mathbf{K}_3 \tanh(\mathbf{K}_4 \tilde{\mathbf{d}}), \quad (2.3)$$

onde $\mathbf{K}_i, i = 1, 2, 3, 4$, são matrizes de ganho diagonais definidas positivas. Uma vez que os termos contendo o erro $\tilde{\mathbf{d}}$ e a derivada do erro $\dot{\tilde{\mathbf{d}}}$ são proporcionais, para simplificar a escolha de ganhos, os ganhos \mathbf{K}_3 e \mathbf{K}_4 foram obtidos a partir de \mathbf{K}_1 e \mathbf{K}_2 com as relações $\mathbf{K}_3 = \sqrt{4\mathbf{K}_1}$ e $\mathbf{K}_4 = \frac{\sqrt{4\mathbf{K}_1\mathbf{K}_2}}{\mathbf{K}_3}$.

² A nomenclatura original do vetor de postura foi modificado aqui para evitar confusões com \mathbf{q} das variáveis da formação apresentado na seção seguinte.

3 Estrutura e controle da formação

Este capítulo descreve as estruturas das formações, começando com o segmento de reta 2D, seguido de sua modificação para o espaço 3D, enfatizando a relação entre as variáveis associadas à formação e as variáveis dos robôs. Em seguida a estratégia de controle proposta é apresentada, finalizando com a análise de estabilidade do controlador.

3.1 Caracterização da formação

3.1.1 Formação 2D

Uma formação de robôs pode assumir diferentes *layouts*, dependendo da quantidade e objetivos da tarefa. No método de estruturas virtuais, a formação é definida como uma forma geométrica, onde um ponto específico será o objeto de controle. Para tanto, é necessário conhecer a relação entre cada elemento e a estrutura formada. Isto pode ser obtido através de análise geométrica, resultando nas relações entre as variáveis da formação e dos robôs. A formação com dois robôs pode ser descrita facilmente por um segmento de reta como na Figura 5.

Os pontos $\mathbf{h}_1 = [x_1 \ y_1]^T$ e $\mathbf{h}_2 = [x_2 \ y_2]^T$ representam os pontos de interesse para o controle de cada um dos robôs.

A forma da formação é descrita pelas variáveis ρ_f , que define a distância que os robôs devem manter entre si, e α_f , que define o ângulo entre a reta que une os dois robôs e o eixo horizontal do sistema de coordenadas global, enquanto a posição da formação é definida pelas coordenadas do centro de gravidade da mesma linha reta, a saber (x_f, y_f) . Portanto, o objetivo de controle é posicionar os robôs tal que os pontos \mathbf{h}_1 e \mathbf{h}_2 cumpram

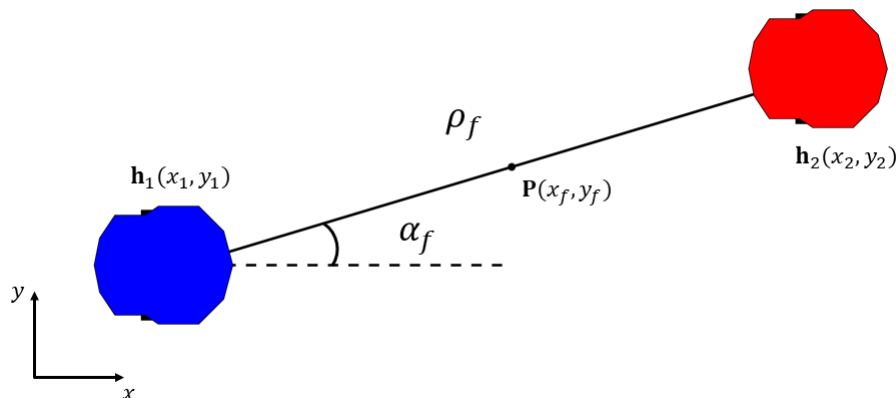


Figura 5 – Configuração da formação como segmento de reta bidimensional.

Fonte: Autoria própria.

os requisitos de forma e posição desejadas da formação. Logo, há uma relação entre as variáveis que descrevem a formação (chamadas variáveis da formação), aqui definidas como

$$\mathbf{q} = [x_f \quad y_f \quad \rho_f \quad \alpha_f]^T,$$

e as posições dos robôs (chamadas variáveis dos robôs), aqui agrupadas no vetor

$$\mathbf{x} = [\mathbf{h}_1^T \quad \mathbf{h}_2^T]^T = [x_1 \quad y_1 \quad x_2 \quad y_2]^T,$$

que são as chamadas transformações do espaço da formação (KITTS; MAS, 2009; BRANDÃO et al., 2015) para o espaço dos robôs e do espaço dos robôs para o espaço da formação. A **transformação direta** obtém as variáveis da formação a partir das variáveis dos robôs, dada pela relação

$$\mathbf{q} = f(\mathbf{x}), \quad (3.1)$$

onde

$$x_f = \frac{x_1 + x_2}{2} \quad (3.2)$$

$$y_f = \frac{y_1 + y_2}{2} \quad (3.3)$$

$$\rho_f = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.4)$$

$$\alpha_f = \operatorname{tg}^{-1} \left(\frac{y_2 - y_1}{x_2 - x_1} \right). \quad (3.5)$$

Já no que se refere à transformação do espaço da formação para o espaço dos robôs tem-se a **transformação inversa**, descrita por

$$\mathbf{x} = f^{-1}(\mathbf{q}), \quad (3.6)$$

onde

$$x_1 = x_f - \frac{\rho_f}{2} \cos(\alpha_f) \quad (3.7)$$

$$y_1 = y_f - \frac{\rho_f}{2} \operatorname{sen}(\alpha_f) \quad (3.8)$$

$$x_2 = x_f + \frac{\rho_f}{2} \cos(\alpha_f) \quad (3.9)$$

$$y_2 = y_f + \frac{\rho_f}{2} \operatorname{sen}(\alpha_f). \quad (3.10)$$

Dessa forma, sabendo a posição instantânea de cada robô, o sistema de controle será capaz de inferir o estado atual da formação, e, com base nisso, determinar como os dois robôs devem se mover para manter a formação durante a navegação.

3.1.2 Formação 3D

Considerando um veículo aéreo e um terrestre, este segmento terá característica tridimensional, conforme o esquema na Figura 6. Neste caso é necessária a inclusão de mais duas variáveis de modo a caracterizar a forma 3D, representada pelo vetor

$$\mathbf{q} = [x_f \ y_f \ z_f \ \rho_f \ \alpha_f \ \beta_f]^T,$$

onde x_f , y_f e z_f são coordenadas do ponto de referência da formação, ρ_f é a distância entre os membros da formação, α_f representa o ângulo entre o eixo x do referencial global e a projeção de ρ_f sobre o plano xy , e β_f é o ângulo entre o plano xy e a posição do robô aéreo. Para simplificar os cálculos, o centro de referência da formação foi escolhido como o *Pioneer 3-DX*. Logo, as coordenadas do ponto de controle da formação e do robô terrestre são as mesmas.

Os pontos $\mathbf{h}_1 = (x_1, y_1, z_1)$ e $\mathbf{h}_2 = (x_2, y_2, z_2)$ representam os pontos de interesse para o controle, associados ao VTNT e ao VANT respectivamente. Eles formam o vetor de variáveis dos robôs \mathbf{x} ,

$$\mathbf{x} = [x_1 \ y_1 \ z_1 \ x_2 \ y_2 \ z_2]^T.$$

O objetivo é controlar a formação para executar tarefas de posicionamento e seguimento de trajetória. Além disso, os robôs devem se mover no sentido de manter a formação ao longo do curso da navegação. Então, é imprescindível obter a relações de transformação entre os dois grupos de variáveis. Assim como no modelo bidimensional tem-se a **transformação direta** e a **transformação inversa**.

A transformação direta é dada por

$$\mathbf{q} = f(\mathbf{x}), \quad (3.11)$$

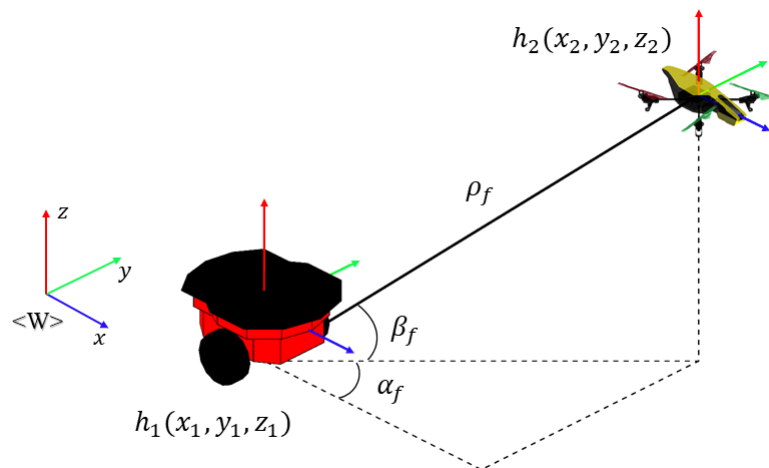


Figura 6 – Configuração da formação como segmento de reta tridimensional.

Fonte: Autoria própria.

com

$$x_f = x_1, \quad (3.12)$$

$$y_f = y_1, \quad (3.13)$$

$$z_f = z_1, \quad (3.14)$$

$$\rho_f = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}, \quad (3.15)$$

$$\alpha_f = \operatorname{tg}^{-1} \left(\frac{y_2 - y_1}{x_2 - x_1} \right), \quad (3.16)$$

$$\beta_f = \operatorname{tg}^{-1} \left(\frac{z_2 - z_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \right). \quad (3.17)$$

Por sua vez, a transformação inversa é

$$\mathbf{x} = f^{-1}(\mathbf{q}), \quad (3.18)$$

onde

$$x_1 = x_f, \quad (3.19)$$

$$y_1 = y_f, \quad (3.20)$$

$$z_1 = z_f, \quad (3.21)$$

$$x_2 = x_f + \rho_f \cos(\alpha_f) \cos(\beta_f), \quad (3.22)$$

$$y_2 = y_f + \rho_f \operatorname{sen}(\alpha_f) \cos(\beta_f), \quad (3.23)$$

$$z_2 = z_f + \rho_f \operatorname{sen}(\beta_f). \quad (3.24)$$

Com isso, todas as relações entre os dois grupos de variáveis foram definidas, e é possível prosseguir para a etapa do controle da formação.

3.2 O controle multicamadas

Para realizar as tarefas propostas, uma estrutura de controle multicamadas é utilizada, similar à descrita em (BRANDÃO et al., 2015; BRANDÃO; SARCINELLI-FILHO, 2016). No entanto será adaptada para dois robôs, como mostra o diagrama na Figura 7. Há duas regiões no modelo. Uma é o Espaço da Formação, contendo a Camada de Planejamento *Off-line*, Planejamento *On-line* e Camada de Controle. A outra é o Espaço dos Robôs, contendo a Camada dos Robôs e a Camada Ambiente. Segue uma breve descrição de cada etapa.

- Camada de Planejamento *Off-line*

O Planejamento *Off-line* ocorre antes de iniciado o controle, e é onde o objetivo da formação é definido. Para controle de posição, as posições finais \mathbf{q}_d são escolhidas. Para o caso de controle de trajetória, o caminho e as respectivas velocidades desejadas $\dot{\mathbf{q}}_d$ são especificadas. Além disso, as posições iniciais dos robôs \mathbf{x}_0 também são fornecidas nesta etapa.

- Camada de Planejamento *On-line*

Ocorre durante o controle, e é responsável por possíveis alterações das variáveis de formação desejada em tempo real, garantindo flexibilidade para a estrutura se for preciso.

- Camada de controle

Na próxima etapa, na Camada de Controle, está presente o controlador responsável por guiar a formação como um todo (como uma estrutura virtual). Neste ponto são geradas as posições de referência \mathbf{q}_r e velocidades de referência da formação $\dot{\mathbf{q}}_r$. Antes de passar para a próxima camada, as referências da formação são convertidas

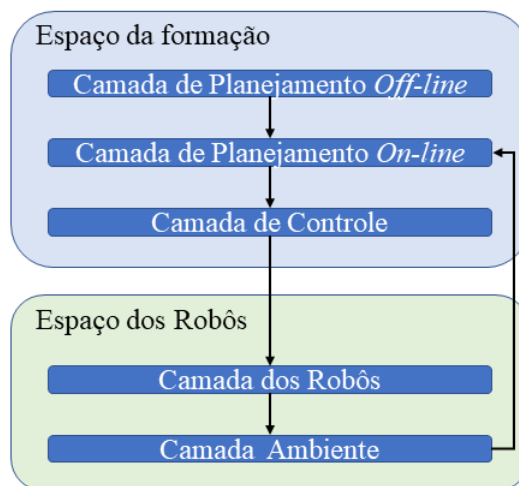


Figura 7 – Estrutura do sistema de controle multicamadas.

Fonte: Autoria própria.

para referências dos robôs através da matriz inversa do Jacobiano (Equações (3.26) e (3.27)) e equações de transformações inversas descritas nas subseções anteriores (Equações (3.6) e (3.18)).

- Camada dos robôs

Nesta camada, os sinais de referências (já convertidos nas variáveis dos robôs) são utilizados para efetuar o controle individual dos robôs de modo a gerar os sinais de controle, conforme explicado no Capítulo 2.

- Camada Ambiente

Aqui, serão obtidas as informações de postura atual através de sensores, sejam eles embarcados ou não. Por isto o nome Camada Ambiente, pois é o resultado da interação entre a formação e o local em que se encontra (e não porque as leituras vêm de sensores do ambiente). Os dados desta camada são utilizados para realimentar os controladores da formação e dos robôs.

Independentemente de ser formação 2D ou 3D, o processo é o mesmo. A diferença está apenas no número de variáveis e no controle individual dos robôs. O diagrama de blocos da Figura 8 funciona para os dois casos.

O controlador (Equação (3.25)) recebe como entrada as posições e velocidades desejadas da formação \mathbf{q}_d e $\dot{\mathbf{q}}_d$ fornecidas pelo usuário, além da posição atual da formação \mathbf{q} . Esta, por sua vez, é calculada aplicando a Transformação Direta (Equações (3.1) e (3.11)) sobre as posições atuais dos robôs \mathbf{x} , obtidas através de sensores (embarcados ou não), pertencentes à Camada Ambiente. Com esses dados, os sinais de referência $\dot{\mathbf{q}}_r$ são calculados.

Continuando o processo, os sinais de saída do controlador $\dot{\mathbf{q}}_r$ são multiplicados pela matriz inversa do Jacobiano resultando nos sinais de referência $\dot{\mathbf{x}}_r$. A próxima etapa consiste no controle de cada robô, porém, para que isto seja possível, as posições desejadas \mathbf{x}_d e atuais \mathbf{x} , além das velocidades atuais $\dot{\mathbf{x}}$ dos robôs devem ser conhecidas. As posições

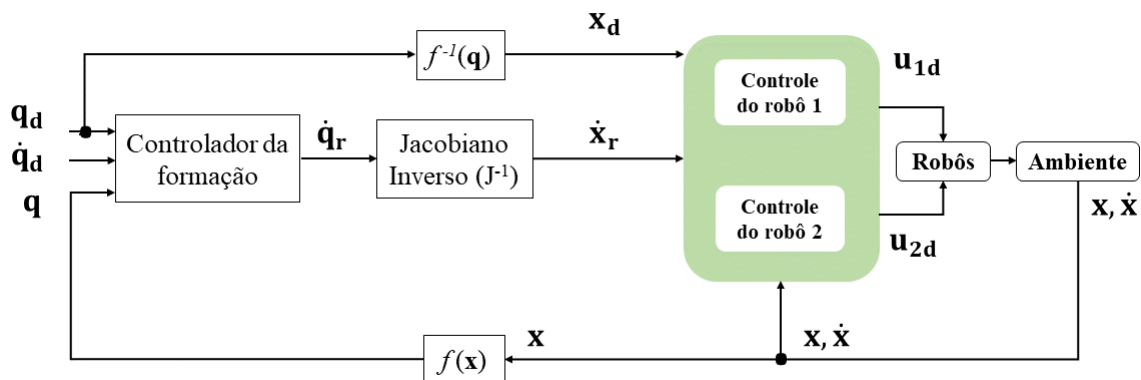


Figura 8 – Diagrama de blocos do sistema de controle.

Fonte: Autoria própria.

desejadas são calculadas com a Transformação Inversa (Equações (3.6) e (3.18)) a partir de \mathbf{q}_d , enquanto \mathbf{x} e $\dot{\mathbf{x}}$ vêm da Camada Ambiente.

O bloco de controle dos robôs segue conforme explicado no Capítulo 2. Para o *Pioneer 3-DX*, esta etapa é realizada passando pelo modelo cinemático inverso, que permite calcular os sinais de controle de referência, seguido de compensação dinâmica. Para o *AR.Drone 2.0*, é utilizado o controlador dinâmico subatuado em conjunto com o modelo dinâmico inverso do VANT. Nos dois casos, as saídas são os sinais de controle \mathbf{u}_d para acionar os atuadores dos veículos. Depois, são realizadas as leituras dos sensores na Camada Ambiente, permitindo a realimentação dos controladores da formação e dos robôs.

3.2.1 Controlador da formação

O controlador proposto é diretamente proporcional ao erro da formação. Desse modo, quanto mais distante do objetivo, maior será o sinal de controle. À medida que a formação se aproxima do alvo, o erro diminui, e conseqüentemente o sinal de controle também. Essa característica evita que o robô se aproxime do destino com velocidade elevada e possivelmente o ultrapasse.

O controlador é baseado no critério de estabilidade de Lyapunov, aplicável a sistemas não lineares. Para tanto, a lei de controle escolhida é

$$\dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d + \mathbf{L}_1 \tanh(\mathbf{L}_2 \tilde{\mathbf{q}}). \quad (3.25)$$

A variável $\dot{\mathbf{q}}_r$ é o sinal de referência da formação e $\dot{\mathbf{q}}_d$ representa as velocidades desejadas. O erro de formação é $\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q}$, \mathbf{L}_1 e \mathbf{L}_2 são matrizes diagonais definidas positivas de ganho. Para a formação 2D, $\{\mathbf{L}_1, \mathbf{L}_2\} \in \mathfrak{R}^4$, e para a formação 3D, $\{\mathbf{L}_1, \mathbf{L}_2\} \in \mathfrak{R}^6$. A função $\tanh(\cdot)$ funciona como um limitador, prevenindo a saturação do sinal de controle em relação aos valores máximos permitidos pelos robôs.

Para o caso de posicionamento, o alvo de destino não varia com o tempo, portanto $\dot{\mathbf{q}}_d = \mathbf{0}$. Para seguimento de trajetória, o objetivo é variável. Logo, $\dot{\mathbf{q}}_d \neq \mathbf{0}$, sendo que \mathbf{q}_d e $\dot{\mathbf{q}}_d$ caracterizam a trajetória a ser seguida. Portanto o controlador proposto pode ser aproveitado para as duas tarefas, sem modificações.

3.2.2 Matriz Inversa do Jacobiano

Após o cálculo de $\dot{\mathbf{q}}_r$ com a Equação (3.25), é necessário determinar como cada robô irá se mover. Isto é feito através da transformação do espaço da formação $\dot{\mathbf{q}}$ para o espaço dos robôs $\dot{\mathbf{x}}$. A matriz de transformação de espaços é nomeada Jacobiano. No caso em questão, é preciso empregar o Jacobiano inverso, dado por

$$\mathbf{J}^{-1}(\mathbf{q}) = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}$$

Devido às diferenças entre as equações e dimensões de cada formação, a Matriz Jacobiano será diferente para cada caso.

- **Formação 2D**

Para a formação bidimensional, a Matriz Jacobiano inversa fica

$$\mathbf{J}^{-1}(\mathbf{q}) = \begin{bmatrix} \frac{\partial x_1}{\partial x_f} & \frac{\partial x_1}{\partial y_f} & \frac{\partial x_1}{\partial \rho_f} & \frac{\partial x_1}{\partial \alpha_f} \\ \frac{\partial y_1}{\partial x_f} & \frac{\partial y_1}{\partial y_f} & \frac{\partial y_1}{\partial \rho_f} & \frac{\partial y_1}{\partial \alpha_f} \\ \frac{\partial x_2}{\partial x_f} & \frac{\partial x_2}{\partial y_f} & \frac{\partial x_2}{\partial \rho_f} & \frac{\partial x_2}{\partial \alpha_f} \\ \frac{\partial y_2}{\partial x_f} & \frac{\partial y_2}{\partial y_f} & \frac{\partial y_2}{\partial \rho_f} & \frac{\partial y_2}{\partial \alpha_f} \end{bmatrix}$$

$$\mathbf{J}^{-1}(\mathbf{q}) = \begin{bmatrix} 1 & 0 & -\frac{\cos(\alpha_f)}{2} & \rho_f \frac{\sin(\alpha_f)}{2} \\ 0 & 1 & -\frac{\sin(\alpha_f)}{2} & -\rho_f \frac{\cos(\alpha_f)}{2} \\ 1 & 0 & \frac{\cos(\alpha_f)}{2} & -\rho_f \frac{\sin(\alpha_f)}{2} \\ 0 & 1 & \frac{\sin(\alpha_f)}{2} & \rho_f \frac{\cos(\alpha_f)}{2} \end{bmatrix}. \quad (3.26)$$

- **Formação 3D**

No caso tridimensional é o mesmo processo, incluindo as demais variáveis da formação:

$$\mathbf{J}^{-1}(\mathbf{q}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & c_{\alpha_f} c_{\beta_f} & -s_{\alpha_f} c_{\beta_f} \rho_f & -c_{\alpha_f} s_{\beta_f} \rho_f \\ 0 & 1 & 0 & s_{\alpha_f} c_{\beta_f} & c_{\alpha_f} c_{\beta_f} \rho_f & -s_{\alpha_f} s_{\beta_f} \rho_f \\ 0 & 0 & 1 & s_{\beta_f} & 0 & c_{\beta_f} \rho_f \end{bmatrix}, \quad (3.27)$$

onde c e s representam $\cos(\cdot)$ e $\sin(\cdot)$, respectivamente.

Multiplicando a matriz inversa do Jacobiano (Equação (3.26) para formação 2D e Equação (3.27) para formação 3D) pelo vetor $\dot{\mathbf{q}}_r$, as velocidades de referência em relação às coordenadas cartesianas $\dot{\mathbf{x}}_r$ são obtidas para cada caso, finalizando a camada de Controle da Formação.

3.3 Análise de estabilidade do controlador

Para realizar a análise de estabilidade do controlador proposto em malha fechada, considera-se que $\dot{\mathbf{q}} = \dot{\mathbf{q}}_r$, que significa que as mudanças da formação são instantâneas.

Esta suposição é possível porque a estrutura virtual não possui dinâmica, diferentemente dos robôs que a compõem.

Então, pode ser escrito, considerando (3.25),

$$\dot{\mathbf{q}}_d + \mathbf{L}_1 \tanh(\mathbf{L}_2 \tilde{\mathbf{q}}) = \dot{\mathbf{q}},$$

ou

$$\dot{\mathbf{q}}_d - \dot{\mathbf{q}} = \dot{\tilde{\mathbf{q}}} = -\mathbf{L}_1 \tanh(\mathbf{L}_2 \tilde{\mathbf{q}}). \quad (3.28)$$

Seja

$$V = \frac{1}{2} \tilde{\mathbf{q}}^T \tilde{\mathbf{q}}$$

a função candidata de Lyapunov para o sistema de controle proposto. Tomando sua derivada temporal e considerando (3.28), obtém-se

$$\dot{V} = \tilde{\mathbf{q}}^T \dot{\tilde{\mathbf{q}}} = -\tilde{\mathbf{q}}^T \mathbf{L}_1 \tanh(\mathbf{L}_2 \tilde{\mathbf{q}}) < 0,$$

de modo que a derivada temporal é negativa para todo valor não nulo de $\tilde{\mathbf{q}}$, pois a função $\tanh(\cdot)$ é ímpar e \mathbf{L}_1 e \mathbf{L}_2 são matrizes diagonais definidas positivas.

Portanto, de acordo com o teorema de Lyapunov, o controle proposto é assintoticamente estável, ou $\tilde{\mathbf{q}} \rightarrow \mathbf{0}$ quando $t \rightarrow \infty$.

4 Resultados

Com o intuito de validar o controlador proposto, foram executados diversos experimentos práticos. Para a realização dos experimentos, os robôs descritos no Capítulo 2, o *Pioneer 3-DX* e o *AR.Drone 2.0* foram utilizados.

Os valores de ganhos cinemáticos e dinâmicos dos controladores dos robôs foram definidos manualmente através de simulações, e posteriormente tiveram que ser reajustados para os experimentos, pois os modelos não são perfeitos. Os parâmetros θ utilizados para o *Pioneer 3-DX* foram os mesmos obtidos em (MARTINS, 2009) através técnicas de identificação. A Tabela 1 contém os parâmetros e ganhos dos controladores dos robôs adotados para os experimentos.

Os resultados foram organizados em duas seções, uma referente à formação bidimensional e outra à formação tridimensional. O programa utilizado para executar as simulações e experimentos foi o MATLAB R2017a, e por questões de praticidade, o controle da formação 2D foi realizado de modo descentralizado ao passo que o da formação 3D foi de modo centralizado. Mais detalhes da metodologia serão explicados adiante.

4.1 Formação 2D

4.1.1 Metodologia

Com a finalidade de comprovar a efetividade do sistema de controle proposto para guiar a formação, foram conduzidos experimentos de posicionamento e seguimento de trajetória. Para tanto, foram utilizados dois robôs *Pioneer 3-DX* sem computador embarcado. Para realizar o controle de modo descentralizado, cada robô foi conectado a um *notebook* através de um adaptador serial/USB. Os computadores foram posicionados sobre os robôs durante os experimentos, e a comunicação foi via rede *wireless*.

Os robôs executaram o mesmo algoritmo, recebendo pela rede apenas as informações de ID (identificação numérica do robô), postura e velocidade do outro agente da formação.

Tabela 1 – Parâmetros dos robôs para os experimentos.

<i>Pioneer 3-DX</i>	
Ponto de controle (a)	15 cm
Parâmetros (θ)	[0,534 0,217 -0,013 0,956 -0,084 1,059]
Ganhos dinâmicos	$\mathbf{K}_{d1} = \text{diag}[0,75 \ 0,75]$ $\mathbf{K}_{d2} = \text{diag}[0,12 \ 0,035]$
<i>AR.Drone 2.0</i>	
Ganhos do controlador	$\mathbf{K}_1 = \text{diag}[0,5 \ 0,6 \ 2 \ 10 \ 8 \ 1]$ $\mathbf{K}_2 = \text{diag}[3 \ 3 \ 15 \ 3 \ 3 \ 4]$

Tabela 2 – Ganhos do controlador da formação 2D.

Ganhos para controle de posição	$\mathbf{L}_1 = \text{diag}[0, 23 \quad 0, 23 \quad 0, 6 \quad 0, 5]$ $\mathbf{L}_2 = \text{diag}[0, 5 \quad 0, 5 \quad 0, 5 \quad 0, 5]$
Ganhos para seguimento de trajetória	$\mathbf{L}_1 = \text{diag}[0, 4 \quad 0, 4 \quad 0, 6 \quad 1, 5]$ $\mathbf{L}_2 = \text{diag}[0, 5 \quad 0, 5 \quad 0, 5 \quad 0, 5]$

Tabela 3 – Parâmetros das variáveis de formação desejadas para controle de posicionamento 2D.

	x_f [m]	y_f [m]	ρ_f [m]	α_f [graus]
D_0	8	0	0,9	-90
D_1	8	0	0,9	-180
D_2	8	0	0,9	90

Os dados de velocidade e pose atuais foram obtidos a partir dos sensores embarcados de cada robô, na taxa de atualização do *Pioneer 3-DX* de 100 ms, e o experimento começava automaticamente após cada robô ter recebido um pacote de informações do outro.

No caso da tarefa de posicionamento, o algoritmo foi executado até o erro da formação ser menor que 5 cm para as posições e distância entre robôs e 3 graus para o ângulo da formação. No seguimento de trajetória, foi definida uma volta e meia do percurso para finalizar o experimento. Os ganhos do controlador da formação ajustados para os experimentos são apresentados na Tabela 2. Os mesmos parâmetros cinemáticos e dinâmicos foram utilizados para os dois robôs.

4.1.2 Controle de Posição

A tarefa executada foi mover os robôs, inicialmente posicionados em $[x_{10} \quad y_{10} \quad \psi_{10}]^T = [0 \text{ m} \quad 0,45 \text{ m} \quad 0^\circ]^T$ e $[x_{20} \quad y_{20} \quad \psi_{20}]^T = [0 \text{ m} \quad -0,45 \text{ m} \quad 0^\circ]^T$ para a formação descrita na Tabela 3, ou seja, mantendo os robôs a 0,9 m um do outro, sendo a reta que os une paralela ao eixo x do sistema de coordenadas global, e com seu centro nas coordenadas (8 m, 0 m).

Para demonstrar que a Camada de Planejamento *On-line* permite alterar as variáveis da formação, o ângulo da formação α_f foi modificado em dois momentos. Esta sequência é útil por exemplo para a passagem da formação através de uma porta estreita¹.

Os valores iniciais são mostrados na linha D_0 da Tabela 3. Quando 25% do percurso total é alcançado, o ângulo varia para o mostrado na linha D_1 . Depois, após 65% do caminho concluído, há mais uma alteração, conforme a linha D_2 . Ou seja, os robôs começam lado a lado, e seguem em paralelo, para então formarem uma fila e depois retornam à configuração inicial, com posições trocadas.

A Figura 9 mostra o caminho percorrido pelos dois robôs até atingir o destino. A formação de fato se posicionou corretamente e também executou as alterações na forma

¹ O vídeo do experimento pode ser acessado em <<https://www.youtube.com/watch?v=vrVrsilt3mk>>.

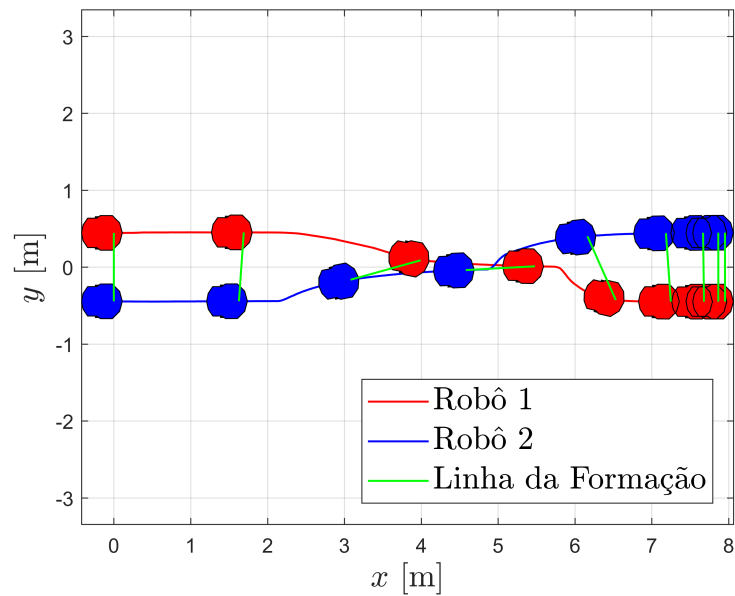
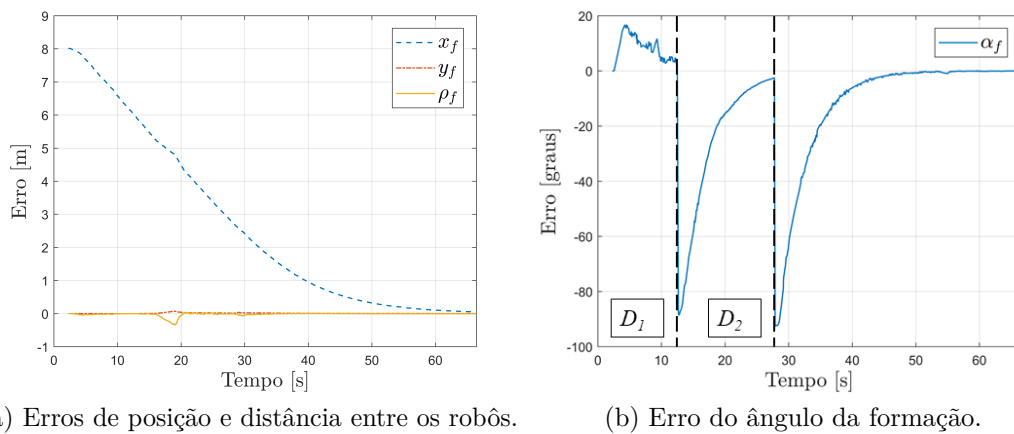


Figura 9 – Percurso realizado para tarefa de posicionamento 2D.



(a) Erros de posição e distância entre os robôs.

(b) Erro do ângulo da formação.

Figura 10 – Evolução temporal dos erros das variáveis da formação.

como desejado, o que pode ser confirmado na Figura 10, onde os erros da formação convergem assintoticamente para zero. O erro do ângulo α_f cresce nos momentos em que o valor desejado é alterado, convergindo novamente para zero em seguida.

Por fim, a Figura 11 mostra as velocidades lineares e angulares desenvolvidas² pelos robôs, confirmando a proporcionalidade do sinal de controle em relação ao erro da formação. Os picos que aparecem nos gráficos ocorrem justamente nos momentos em que o erro é maior, no início do experimento, e nos instantes em que o ângulo da formação muda bruscamente. Em todo o percurso os sinais desejados foram alcançados satisfatoriamente.

² Nos gráficos, os índices d foram utilizados para desejados e r para realizados.

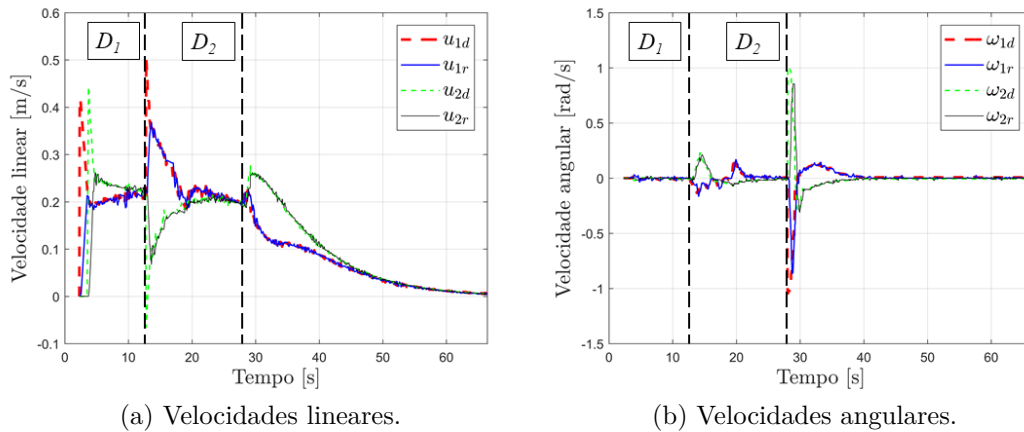


Figura 11 – Evolução temporal das velocidades dos robôs.

Tabela 4 – Parâmetros das variáveis de formação desejadas para trajetória 2D circular.

x_f [m]	y_f [m]	ρ_f [m]	α_f [graus]
$2 \cos(0,15t)$	$2 \sin(0,15t)$	1	$0,15t$

4.1.3 Seguimento de trajetória circular

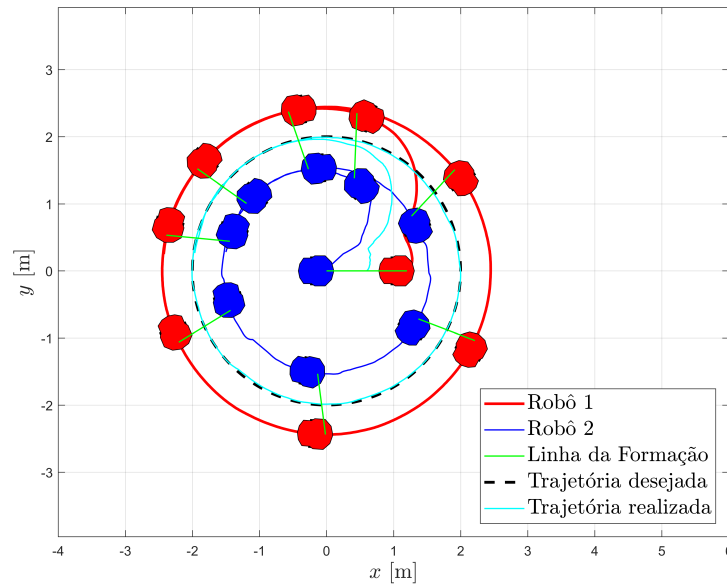
Adicionalmente, tal sistema também pode guiar a formação em tarefas de seguimento de trajetória, bastando para isso especificar ambos os vetores \mathbf{q}_d e $\dot{\mathbf{q}}_d$ em (3.25), como mencionado anteriormente. Em outras palavras, \mathbf{q}_d é definido como uma função do tempo, e $\dot{\mathbf{q}}_d$ é a sua derivada temporal. No experimento em questão foi definida uma trajetória circular³.

As variáveis de formação desejadas são mostradas na Tabela 4. As posições iniciais dos robôs são $\mathbf{h}_1 = (1,2 \text{ m}, 0 \text{ m})$ e $\mathbf{h}_2 = (0 \text{ m}, 0 \text{ m})$. A duração do experimento corresponde a uma volta e meia na circunferência. Os ganhos da formação estão descritos na Tabela 2. A Figura 12a mostra os caminhos de cada um dos robôs e o trajeto do ponto de referência da formação. E as Figuras 12b e 12c ilustram a evolução temporal desejada e efetiva das posições dos robôs.

Na Figura 13 é possível observar o comportamento dos erros relativos às variáveis de formação. Todos se mantiveram consistentemente dentro de uma faixa fixa de valores, próximos de zero.

A Figura 14 mostra as velocidades lineares e angulares comandadas e efetivamente desenvolvidas pelos dois robôs. Analisando as referidas figuras pode-se perceber que o sistema foi capaz de atingir os valores desejados, realizando a trajetória desejada com uma margem de erro pequena.

³ O vídeo do experimento pode ser acessado em <https://www.youtube.com/watch?v=1C8K0g_RiyE>.



(a) Percurso dos robôs.

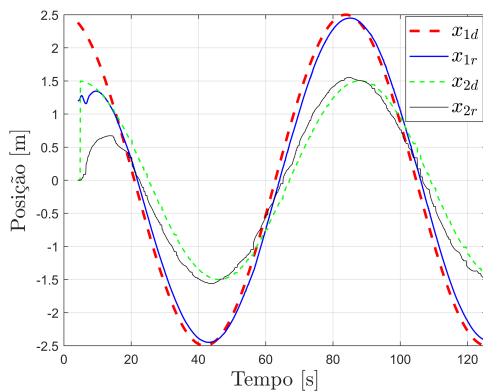
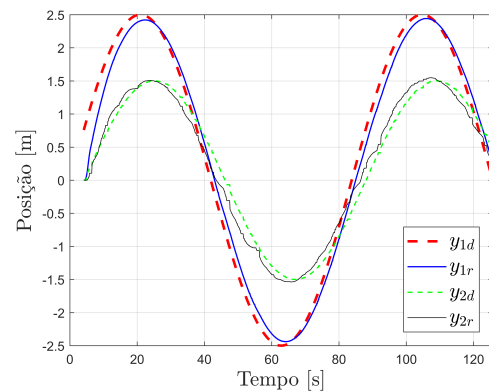
(b) Posições em relação ao eixo x .(c) Posições em relação ao eixo y .

Figura 12 – Percursos dos robôs e do ponto de referência da formação no seguimento da trajetória circular 2D.

4.2 Formação 3D

4.2.1 Metodologia

Para a formação 3D foi utilizado o controle centralizado. O quadrimotor foi conectado ao computador central via wi-fi, e o *Pioneer 3-DX* foi conectado a um *notebook*, da mesma forma que nos experimentos anteriores. Mas nesse caso, o computador conectado ao VTNT não executava nenhum cálculo de controle, apenas recebia os comandos vindo do computador central pela rede *wireless*, e os repassava integralmente ao robô.

Outra modificação para a formação 3D foi a utilização do *OptiTrack* para obtenção da pose dos robôs ao invés de captar as leituras dos sensores embarcados (ver Apêndice A para informações referentes ao *OptiTrack*). As velocidades, por sua vez, foram calculadas derivando a pose.

As quatro câmeras do *OptiTrack* foram conectadas via Ethernet a um *Switch* (do próprio equipamento) e este por sua vez conectado também por Ethernet diretamente no *desktop*, definido como central de controle. Devido ao posicionamento das câmeras, os experimentos foram restringidos ao volume de captura de, aproximadamente, 4 m x 2 m x 2 m. O ciclo de operação do programa foi definido no mesmo valor da taxa de atualização do *AR.Drone 2.0*, de 30 ms, e a frequência de aquisição de imagens das câmeras foi de 120 ou 240 FPS⁴. Como a aquisição de imagens é mais rápida que o ciclo de operação do programa, não houve problemas de atraso na obtenção das posições dos robôs.

Dentre os experimentos executados para esta formação, quatro deles são descritos nas seções seguintes. A Tabela 5 contém os ganhos do controlador da formação utilizados para controle de posicionamento e seguimento de trajetória.

⁴ A taxa de aquisição de imagens e outras configurações do programa variaram de acordo com o experimento, em função de modificações nas configurações do programa para adaptação do sistema às diferentes condições de iluminação do ambiente.

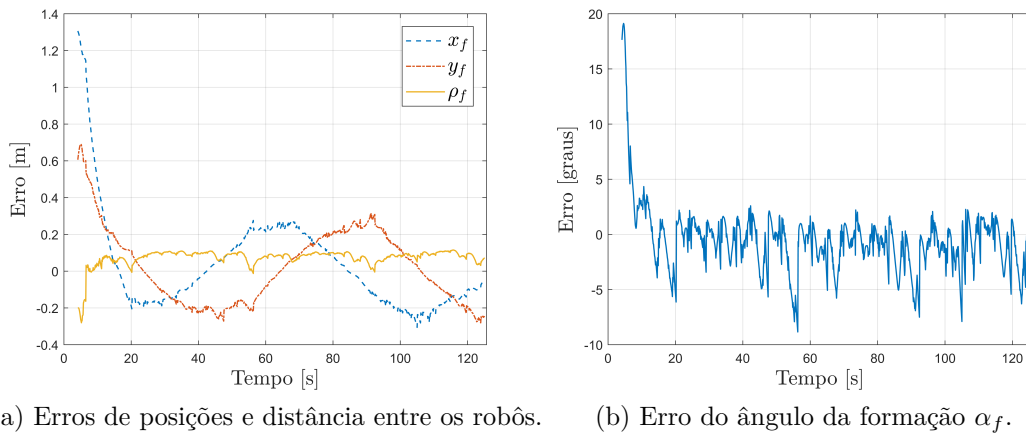


Figura 13 – Erros associados às variáveis de formação durante o seguimento da trajetória 2D circular.

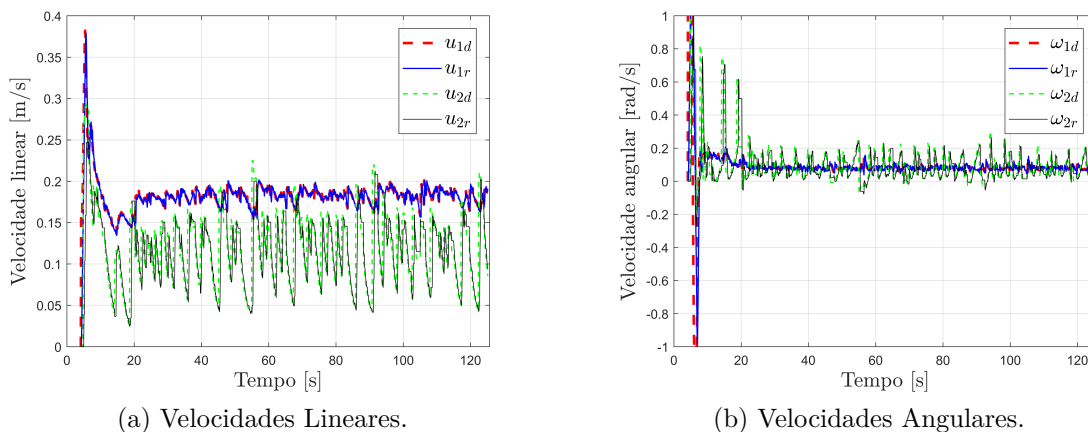


Figura 14 – Velocidades desejadas e efetivamente desenvolvidas pelos robôs da formação.

Tabela 5 – Ganhos do controlador da formação 3D.

Ganhos para controle de posição	$\mathbf{L}_1 = \text{diag}[0,4 \quad 0,4 \quad 0,5 \quad 1,5 \quad 0,75 \quad 0,6]$
	$\mathbf{L}_2 = \text{diag}[0,2 \quad 0,2 \quad 0,5 \quad 0,82 \quad 0,6 \quad 1,57]$
Ganhos para seguimento de trajetória	$\mathbf{L}_1 = \text{diag}[0,6 \quad 0,6 \quad 0,5 \quad 1,2 \quad 0,75 \quad 0,5]$
	$\mathbf{L}_2 = \text{diag}[0,5 \quad 0,5 \quad 0,5 \quad 0,82 \quad 0,6 \quad 1,57]$

Tabela 6 – Variáveis de formação desejada para o experimento de posicionamento 3D.

	x_f [m]	y_f [m]	z_f [m]	ρ_f [m]	α_f [graus]	β_f [graus]
P_1	1,5	0	0	1,5	0	90
P_2	0	1	0	1	0	90
P_3	-1,1	0,5	0	0,6	0	90
P_4	-1,1	-1	0	1	0	90
P_5	0	-1	0	1	0	90
P_6	1	-0,5	0	1	0	90
P_7	-0,5	0	0	$1,5 - (t - t_{total})/5$	0	90

4.2.2 Controle de Posição

Para o controle de posição, foram escolhidos diferentes pontos os quais a formação deveria atingir. A troca de um ponto desejado ao seguinte foi realizada a cada 15 segundos. Quando os robôs alcançam o objetivo antes do tempo definido, eles devem manter sua posição até que o objetivo seja redefinido. As posições desejadas estão descritas na Tabela 6.

Para demonstrar melhor a ideia de cooperação, foi realizado um experimento no qual o VANT aterrissa sobre o VTNT, e tempos depois decola novamente, retornando à forma desejada para continuar a navegação⁵. Esta situação poderia ser útil por exemplo para economia de energia. Imagine uma missão de entrega de suprimentos em ambiente desconhecido. Podem existir obstáculos que dificultem o planejamento do caminho da formação. Nesses casos, imagens aéreas do local são muito úteis. Mas como usualmente o *drone* possui menor tempo de autonomia, ele poderia ser transportado pelo robô terrestre, decolando apenas em momentos de necessidade.

No experimento, o quadrimotor foi programado para aterrissar no ponto P_3 , e decolar novamente em P_5 . Na última posição o VANT aterrissa mais uma vez para finalizar o experimento. A Figura 15 mostra *frames* dos momentos de aterrissagem do *AR.Drone 2.0* sobre a plataforma no ponto P_3 . A plataforma de pouso neste instante encontra-se estática, pois chegou ao objetivo antes do tempo máximo para troca de posições.

Os gráficos dos percursos desejados e realizados no experimento são mostrados em diferentes vistas na Figura 16 para facilitar a visualização. As posições dos robôs e da formação, erros da formação e sinais de controle seguem nas Figuras 17 a 20.

Analisando os gráficos da Figura 17 percebe-se que ambos robôs atingiram as posições desejadas dentro do tempo estabelecido. Em aproximadamente 45 segundos de experimento ocorre a aterrissagem do *drone* sobre o robô terrestre. Como as posições em x e y dos

⁵ O vídeo do experimento pode ser acessado em <<https://www.youtube.com/watch?v=sMEPLpcKCHs>>.

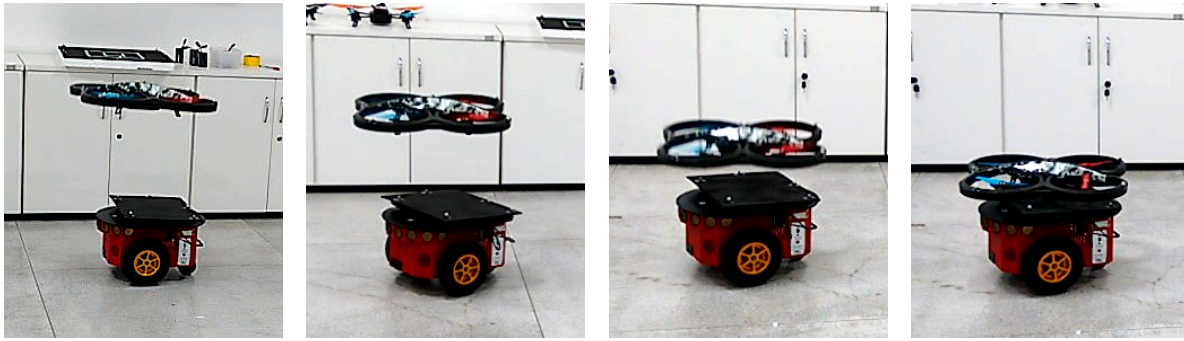


Figura 15 – Sequência de quadros do momento da aterrissagem no experimento de posicionamento.

Fonte: Autoria própria.

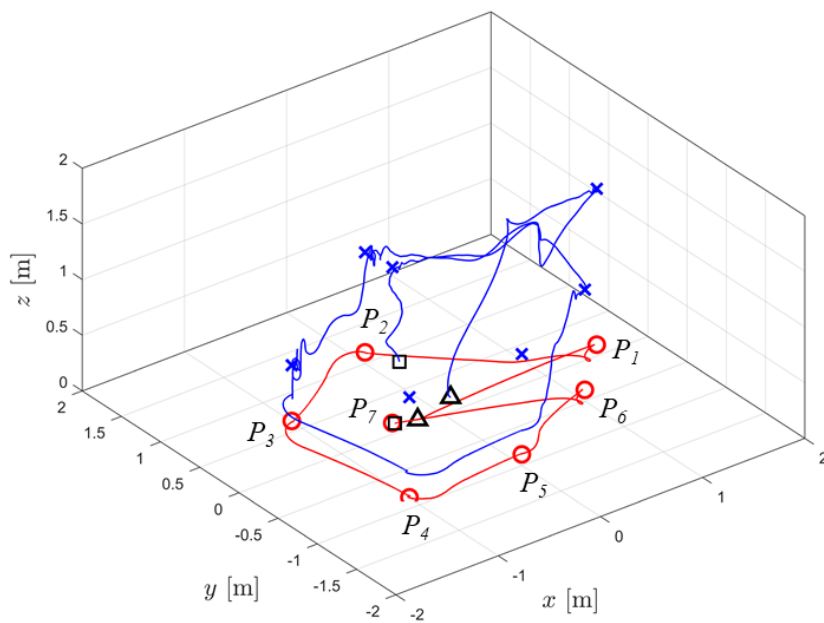
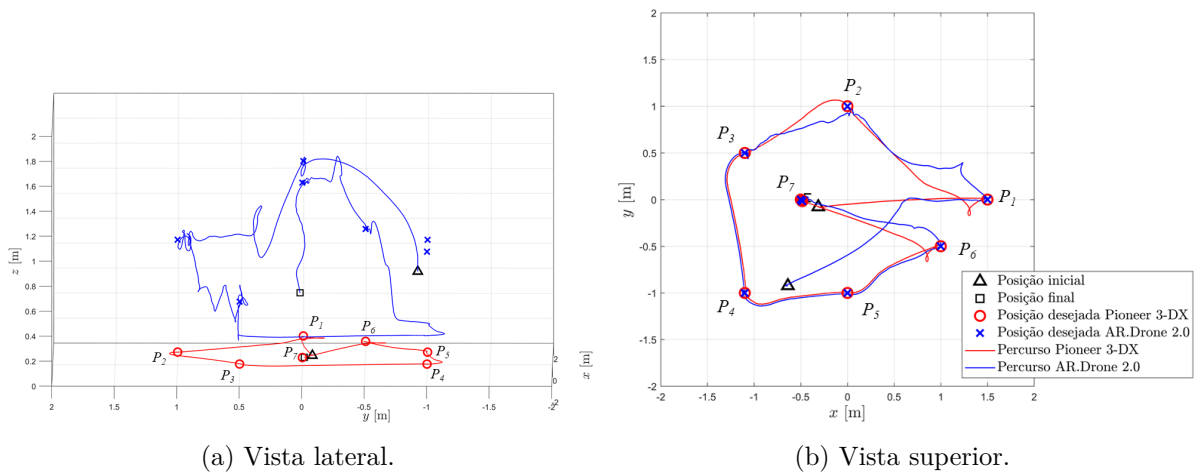
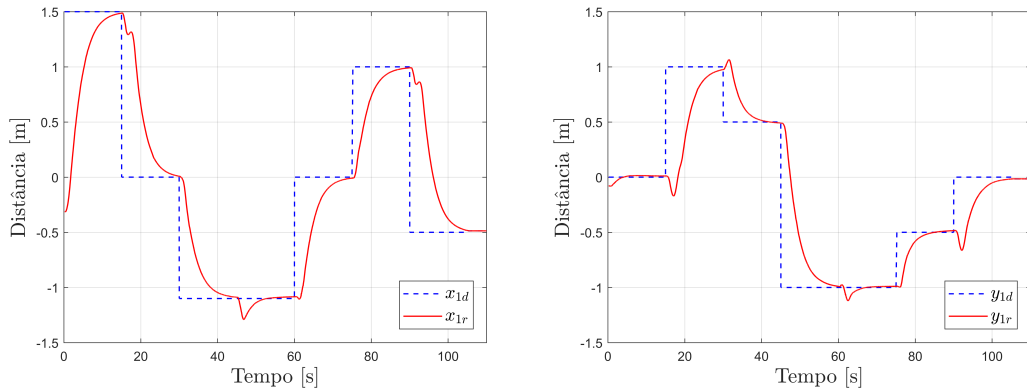
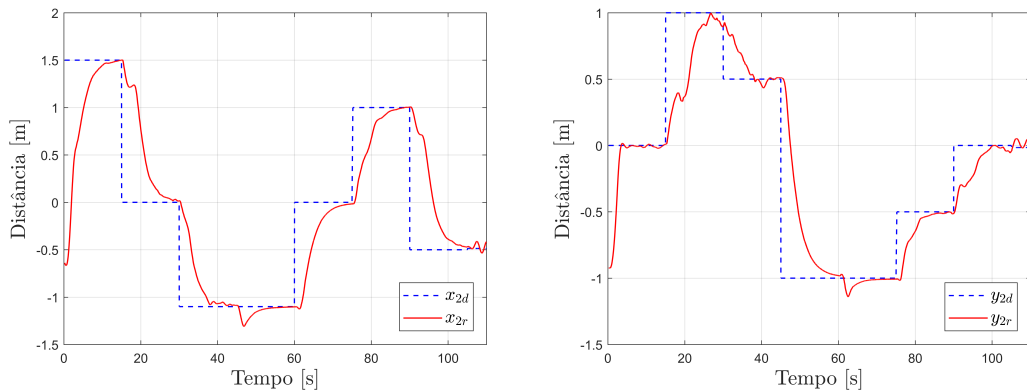


Figura 16 – Vistas do experimento de posicionamento para formação heterogênea.

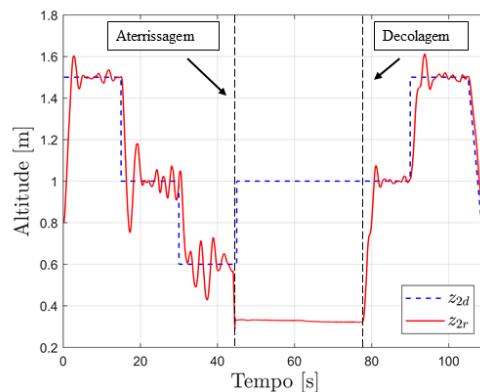
dois robôs são coincidentes, o quadrimotor continua alcançando as posições desejadas, como mostrado nas Figuras 17c e 17d. No intervalo entre pouso e decolagem, claramente a altura desejada não foi alcançada porque o VANT encontrava-se em repouso sobre a plataforma (Figura 17e). Ainda em relação ao eixo z , o *AR.Drone 2.0* demonstrou certa oscilação quando sua altura é alterada instantaneamente. Os resultados demonstram maior



(a) Variação da posição em x do *Pioneer 3-DX*. (b) Variação da posição em y do *Pioneer 3-DX*.



(c) Variação da posição em x do *AR.Drone 2.0*. (d) Variação da posição em y do *AR.Drone 2.0*.



(e) Variação da posição em z do *AR.Drone 2.0*.

Figura 17 – Posições desejadas e realizadas dos robôs ao longo do experimento de posicionamento 3D.

tempo necessário para estabilização quando a altura é reduzida. Estas oscilações afetaram diretamente as variáveis da formação ρ_f e β_f . Este comportamento é refletido nos gráficos da Figura 18.

Quanto às variáveis da formação, como o ponto de controle da formação é o próprio *Pioneer 3-DX*, os gráficos referentes às variáveis $[x \ y \ z]^T$ são os mesmos das Figuras 17a e 17b e por isso não foram repetidos⁶.

As variáveis restantes referem-se às características tridimensionais da formação, e são mostradas na Figura 18. Elas sofrem maior influência do deslocamento do *AR.Drone 2.0*. O gráfico de ρ_f (Figura 18a) por exemplo é praticamente igual à variação de altitude do VANT, na Figura 17e. Este comportamento está dentro do esperado, pois apenas o *drone* varia sua posição no eixo z . O ângulo α_f na Figura 18b oscila bastante, primeiramente devido ao seu intervalo de variação ser $[-180^\circ, 180^\circ]$. Logo há descontinuidades no gráfico quando o valor é menor do que -180° ou passa de 180° . Além disso, quando o ângulo β_f é fixado em 90° , α_f atinge um ponto de singularidade, visto que pode assumir qualquer valor. Isso explica porquê esta variável não segue os valores desejados neste caso. Mesmo assim,

⁶ Em todos os experimentos foi considerada uma superfície plana para navegação do robô terrestre. Portanto, os resultados referentes ao eixo z do *Pioneer 3-DX* foram omitidos, pois são sempre constantes.

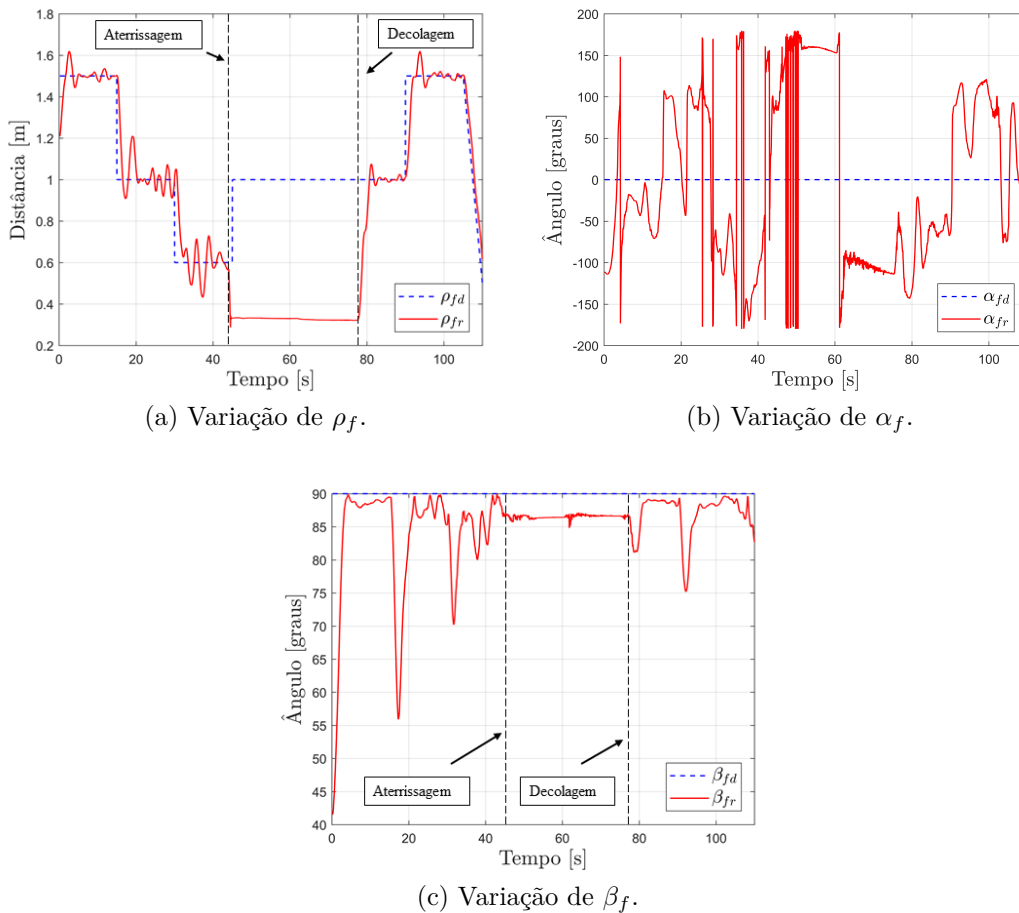


Figura 18 – Variáveis da formação ao longo do percurso.

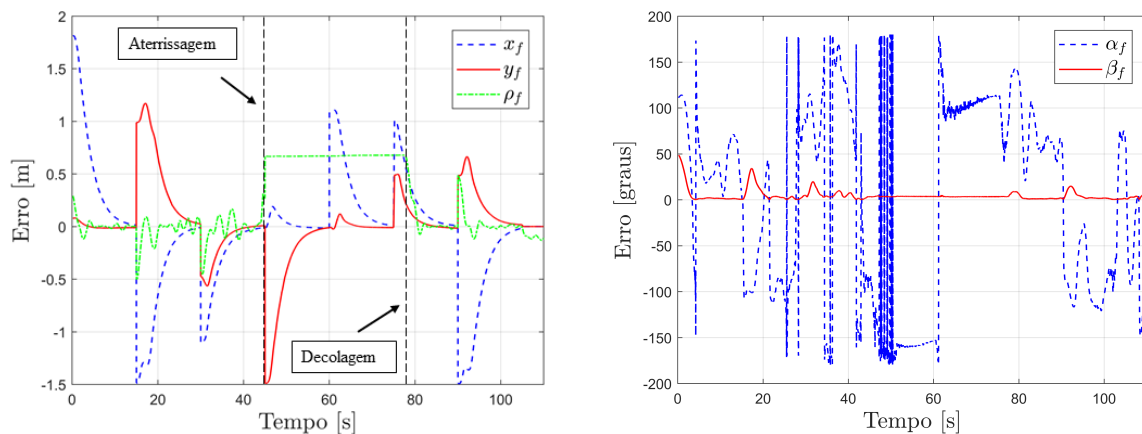
a formação não é perceptivelmente afetada em sua navegação. O ângulo β_f na Figura 18c se aproxima do desejado ao longo do percurso, com maiores erros durante a transição de altitude do quadrimotor. No intervalo entre aterrissagem e decolagem β_f permanece constante, em aproximadamente 87° , ou seja, um erro de apenas 3° .

Os erros das variáveis da formação são exibidos na Figura 19. Com exceção de ρ_f (no intervalo entre aterrissagem e decolagem) e α_f (pelos motivos explicados anteriormente), os erros comportam-se de forma semelhante: iniciam com valores elevados sempre que um novo objetivo é definido, e reduzem assintoticamente com a aproximação do alvo.

Os sinais de controle são diretamente proporcionais ao erro. É por isso que iniciam com altos valores e então reduzem à medida que a formação se aproxima do objetivo. Esta característica pode ser melhor visualizada nas Figuras 20a e 20b devido à forma de movimentação do robô terrestre, que está sempre orientado em relação ao seu movimento. Para o VANT, não foi definido para este experimento taxa de guinada uma vez que ele pode navegar em qualquer direção sem estar orientado em relação ao movimento. No intervalo em que o VANT permanece sobre o robô terrestre, apenas a taxa de guinada apresenta valores realizados diferentes de zero pois é a única que é alterada com a movimentação do VTNT.

4.2.3 Seguimento de Trajetória

Para demonstrar o seguimento de trajetória, foi preciso definir primeiramente um caminho a ser seguido. A velocidade da formação para cada ponto foi obtida derivando-se o percurso desejado. As formas escolhidas foram a Lemniscata de Bernoulli e a Elipse. Para a segunda trajetória, são apresentados resultados de dois experimentos, um com mudança nas variáveis da formação durante o percurso e o outro com aterrissagem e decolagem sobre a plataforma em movimento.



(a) Erros das variáveis da formação.

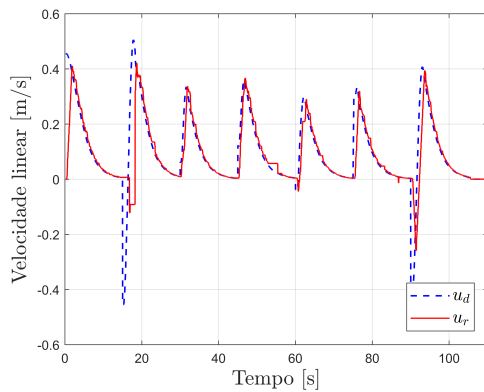
(b) Erros dos ângulos da formação.

Figura 19 – Erros das variáveis de formação para tarefa de posicionamento 3D.

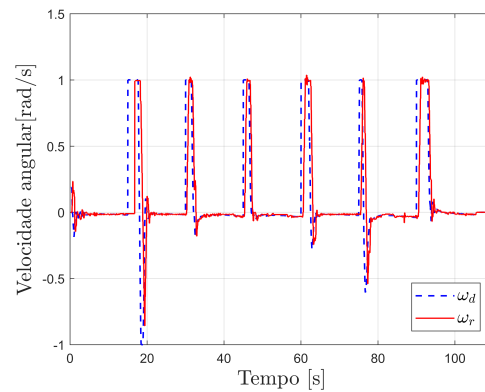
4.2.3.1 Lemniscata de Bernoulli

No primeiro experimento a trajetória escolhida foi a Lemniscata de Bernoulli⁷, por ser capaz de excitar amplamente as velocidades linear e angular, funcionando desse modo como

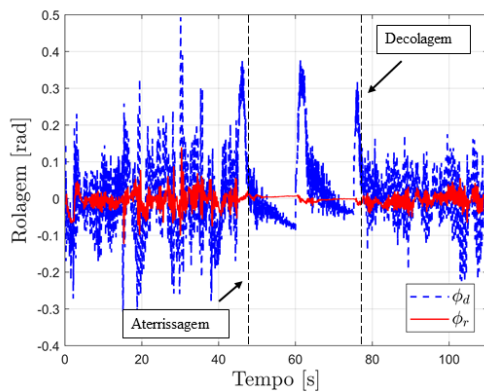
⁷ O vídeo deste experimento pode ser acessado em <https://www.youtube.com/watch?v=B4OoXBxFj0U>.



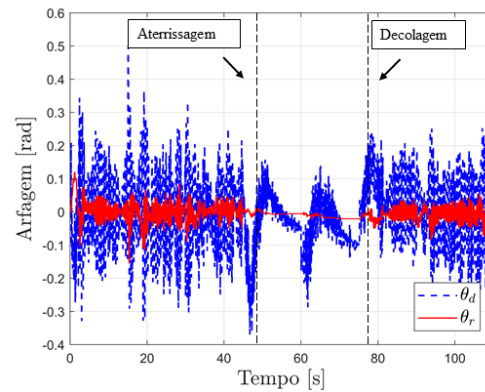
(a) Sinal de controle v do *Pioneer 3-DX*.



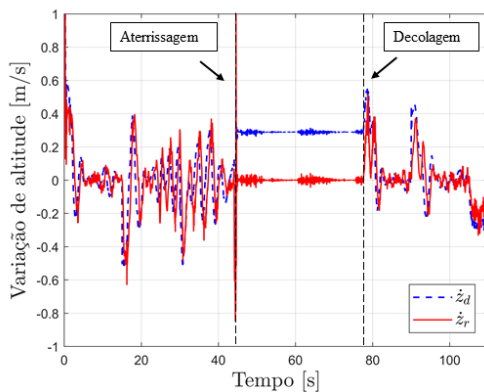
(b) Sinal de controle ω do *Pioneer 3-DX*.



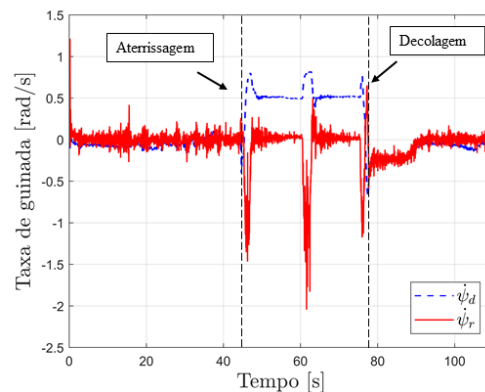
(c) Sinal de controle ϕ do *AR.Drone 2.0*.



(d) Sinal de controle θ do *AR.Drone 2.0*.



(e) Sinal de controle \dot{z} do *AR.Drone 2.0*.



(f) Sinal de controle $\dot{\psi}$ do *AR.Drone 2.0*.

Figura 20 – Sinais de controle desejados e realizados dos robôs ao longo do experimento de posicionamento 3D.

Tabela 7 – Parâmetros das variáveis de formação desejada para a Lemniscata.

x_f [m]	y_f [m]	z_f [m]	ρ_f [m]	α_f [graus]	β_f [graus]
$1,1 \text{ sen}(0,15t)$	$0,8 \text{ sen}(0,3t)$	0	1,2	0	90

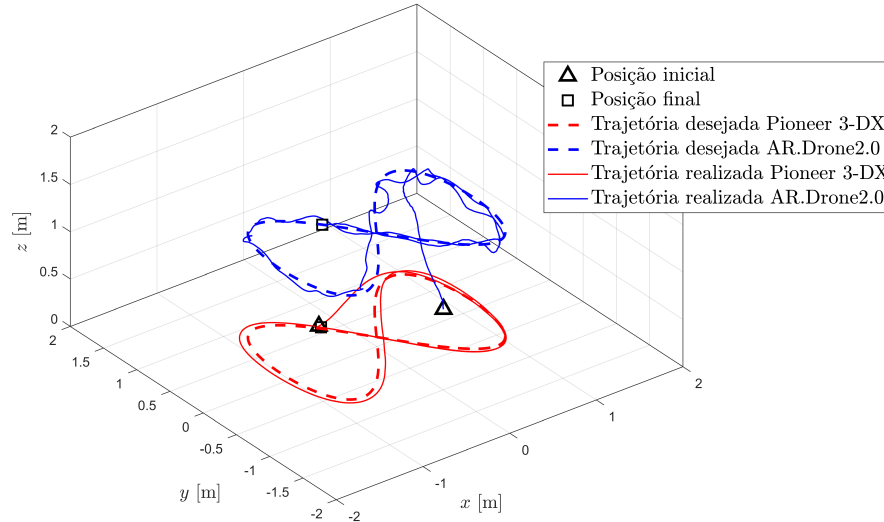


Figura 21 – Trajetória Lemniscata desejada e realizada pela formação.

um bom teste para o controle. Os valores das variáveis da formação para este experimento são apresentados na Tabela 7, e o gráfico com o percurso realizado é exibido na Figura 21.

A Figura 22 apresenta as variações das posições desejadas e realizadas pelos robôs. Os dois veículos conseguem atingir as posições desejadas, sendo o maior desvio de $\pm 10\text{cm}$ do VANT no eixo z . A distância entre os robôs, e os ângulos da formação estão na Figura 23, e os erros das variáveis da formação na Figura 24.

Os erros referentes à posição e ao comprimento da linha mantiveram-se também em uma faixa de $\pm 10\text{cm}$. Quanto aos ângulos da formação, β_f teve um erro próximo de zero enquanto α_f teve comportamento oscilatório (cujas causas foram discutidas no experimento anterior).

Os sinais de controle podem ser vistos na Figura 25. É possível perceber nas Figuras 25a e 25b pontos onde houveram perdas de comunicação entre o robô terrestre e a central. Ainda assim, não afetaram o desempenho do sistema. Para este experimento foi definida uma taxa de guinada, de modo que o VANT estivesse alinhado com seu movimento. A Figura 25f mostra que ele foi capaz de seguir os movimentos desejados.

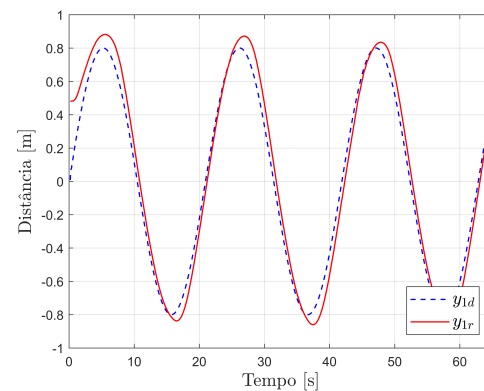
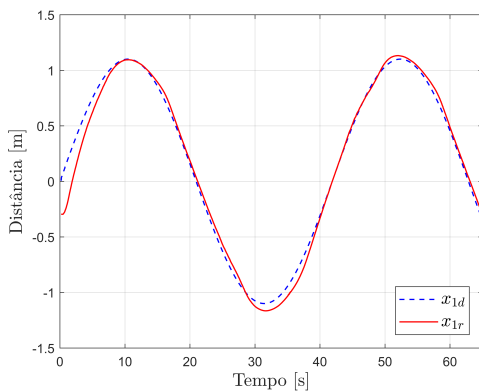
4.2.3.2 Elipse

Outro experimento realizado foi a trajetória elíptica⁸. A formação realizou duas voltas completas, e ao fim da segunda volta, retornou ao centro para finalizar o experimento com o VANT aterrissando sobre o robô terrestre. Na segunda volta algumas variáveis

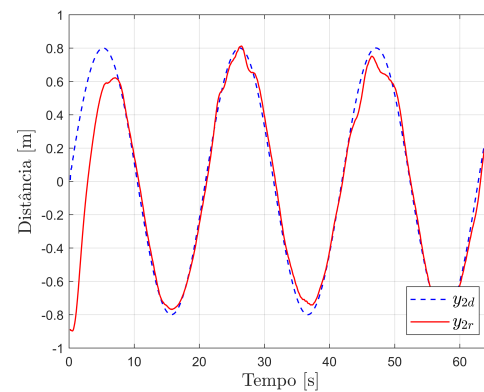
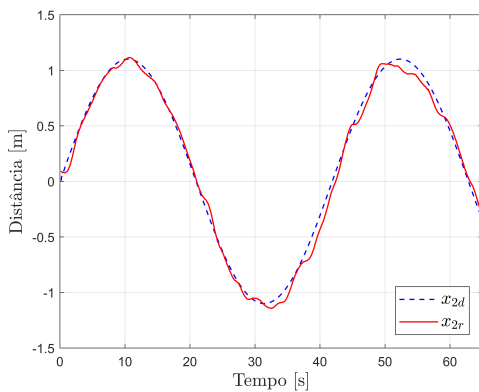
⁸ O vídeo do experimento pode ser acessado em <https://www.youtube.com/watch?v=R4bK2y4RdNw>.

da formação desejadas foram modificadas para demonstrar a atuação da Camada de Planejamento *On-line* do Controle Multicamadas.

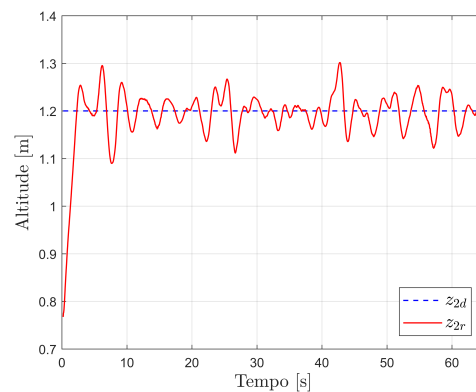
A Tabela 8 contém as informações referentes a cada volta e da aterrissagem ao final da terceira volta. Com intuito de suavizar a aterrissagem do VANT na plataforma foi definida uma função para reduzir a altura gradativamente antes do pouso. A constante t_{total} é o tempo de execução das duas voltas completas. Quanto ao ângulo da formação



(a) Variação da posição em x do *Pioneer 3-DX*. (b) Variação da posição em y do *Pioneer 3-DX*.



(c) Variação da posição em x do *AR.Drone 2.0*. (d) Variação da posição em y do *AR.Drone 2.0*.



(e) Variação da posição em z do *AR.Drone 2.0*.

Figura 22 – Posições desejadas e realizadas dos robôs para a trajetória Lemniscata.

α_f , foi somado o ângulo de guinada do robô terrestre ψ_1 . Isto porque α_f é calculado em relação ao eixo de coordenadas global. Logo, para que o ângulo seja em relação ao robô,

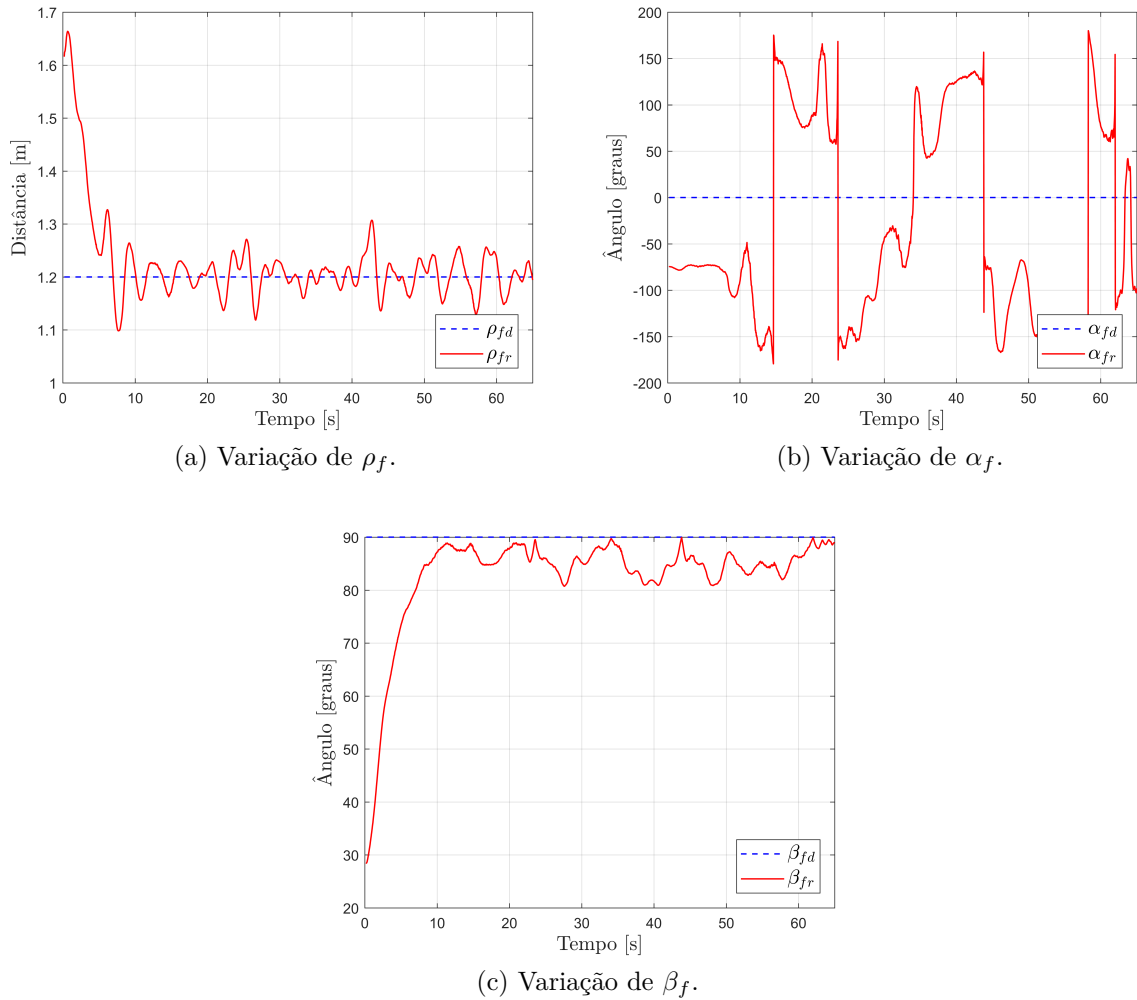


Figura 23 – Variáveis da formação ao longo da trajetória Lemniscata.

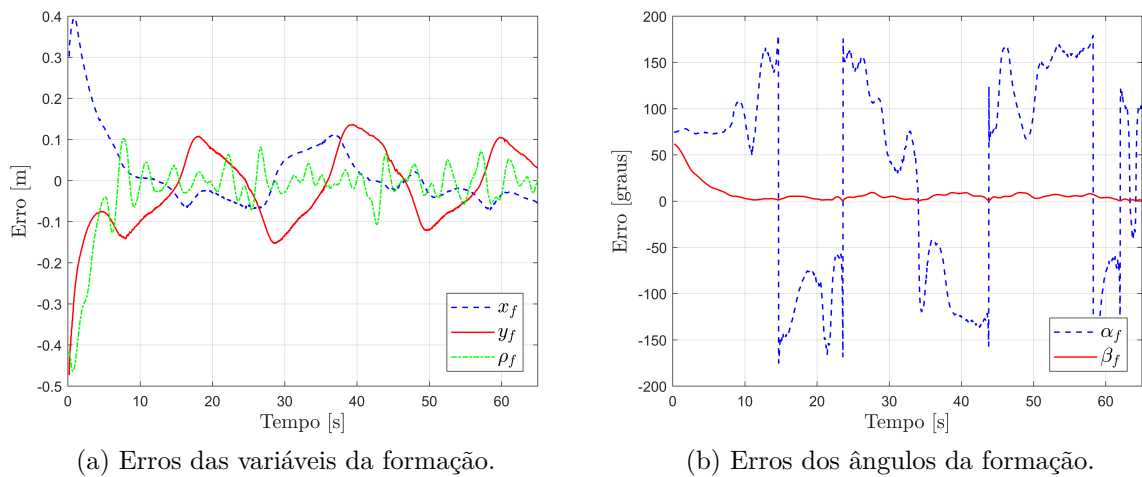


Figura 24 – Erros das variáveis de formação para trajetória Lemniscata.

Tabela 8 – Parâmetros das variáveis de formação desejada para a elipse.

	x_f [m]	y_f [m]	z_f [m]	ρ_f [m]	α_f [graus]	β_f [graus]
1ª volta	$\cos(0,15t)$	$0,5 \sin(0,15t)$	0	1,5	$-90 + \psi_1$	60
2ª volta	$1,5 \cos(0,15t)$	$\sin(0,15t)$	0	1,5	$90 + \psi_1$	60
Pouso	0	0	0	$1,5 - (t - t_{total})/15$	$90 + \psi_1$	90

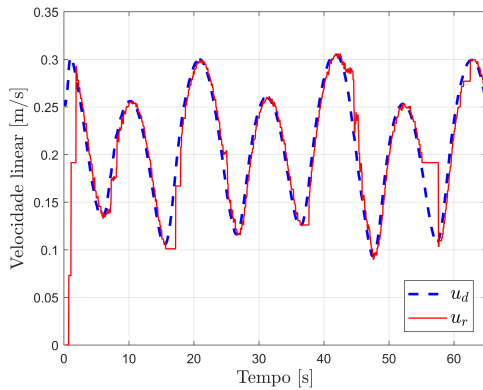
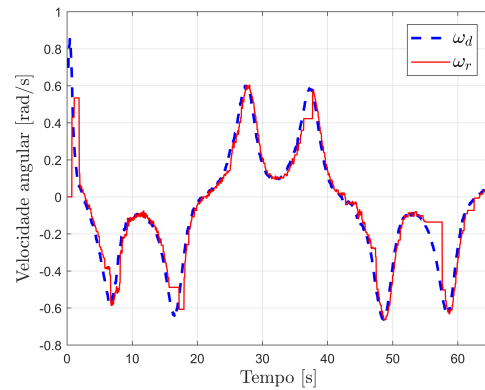
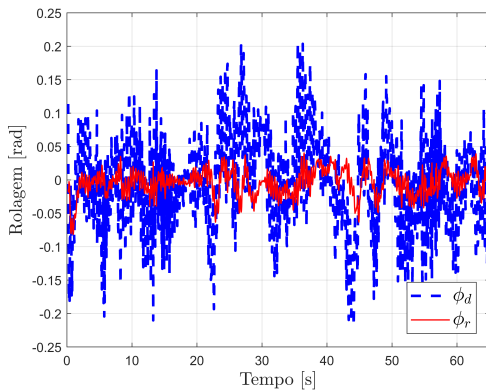
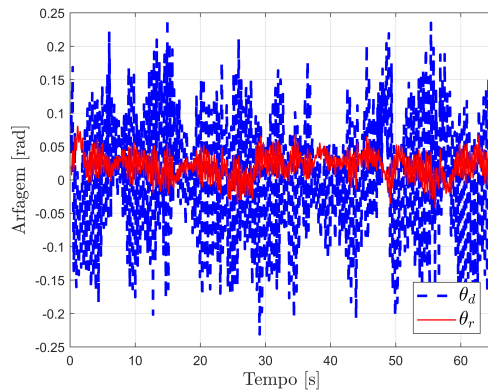
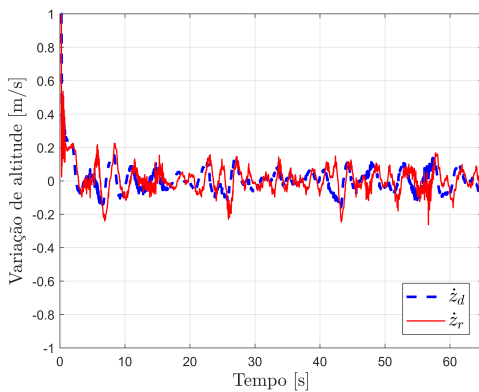
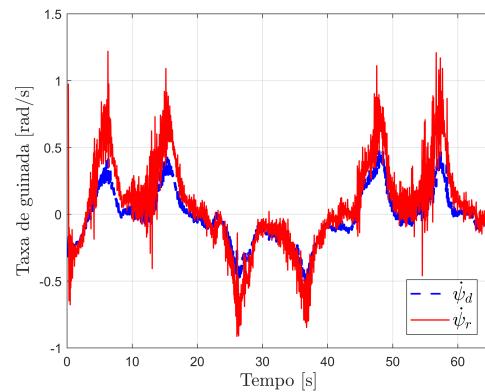
(a) Sinal de controle u do *Pioneer 3-DX*.(b) Sinal de controle ω do *Pioneer 3-DX*.(c) Sinal de controle ϕ do *AR.Drone 2.0*.(d) Sinal de controle θ do *AR.Drone 2.0*.(e) Sinal de controle \dot{z} do *AR.Drone 2.0*.(f) Sinal de controle $\dot{\psi}$ do *AR.Drone 2.0*.

Figura 25 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória Lemniscata.

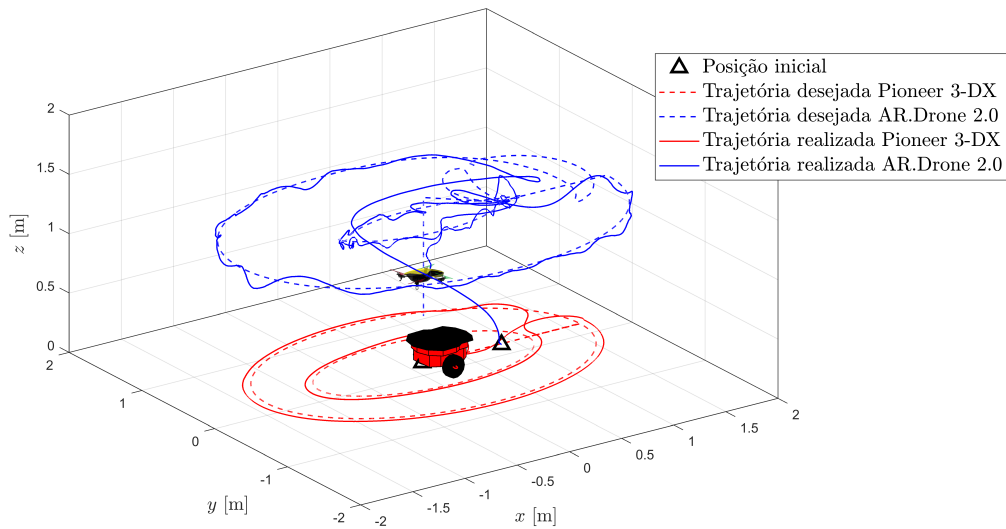


Figura 26 – Trajetória Elíptica desejada e realizada pela formação.

deve-se somar o ângulo de guinada do *Pioneer 3DX* para realizar essa troca dos eixos de referência.⁹

A Figura 26 representa o percurso completo de cada robô durante o experimento. Para facilitar a visualização, o gráfico também foi separado em duas partes, a primeira com um pouco mais de uma volta completa percorrida pela formação, e a segunda mostra o percurso restante. As duas partes são ilustradas na Figura 27.

No momento de transição que ocorre da primeira para a segunda volta, o VANT não consegue alcançar as posições desejadas imediatamente, conforme pode ser percebido na Figura 27b. Esse desvio pode ser visualizado também nas regiões destacadas da Figura 28. No tempo de aproximadamente 41 segundos há uma mudança brusca nas posições desejadas, que não são acompanhadas instantaneamente pelos robôs, o que era esperado considerando os efeitos de inércia sobre eles. Porém, em menos de 10 segundos os robôs voltam a acompanhar a trajetória desejada.

A Figura 29 mostra a evolução das variáveis da formação ao longo da trajetória. Nos momentos de troca das variáveis de formação ocorre uma maior oscilação nos resultados. A Figura 30 contém os erros da formação durante o experimento. Os maiores erros, conforme esperado, acontecem nos pontos em que a formação é alterada e os robôs devem adequar-se à nova configuração. Fora desses pontos os erros se mantêm próximos de zero. Os gráficos dos sinais de controle são mostrados Figura 31.

⁹ No caso de $\beta_f = 90^\circ$, este ajuste é desnecessário. Por isso ψ_1 foi omitido na tabela do experimento anterior.

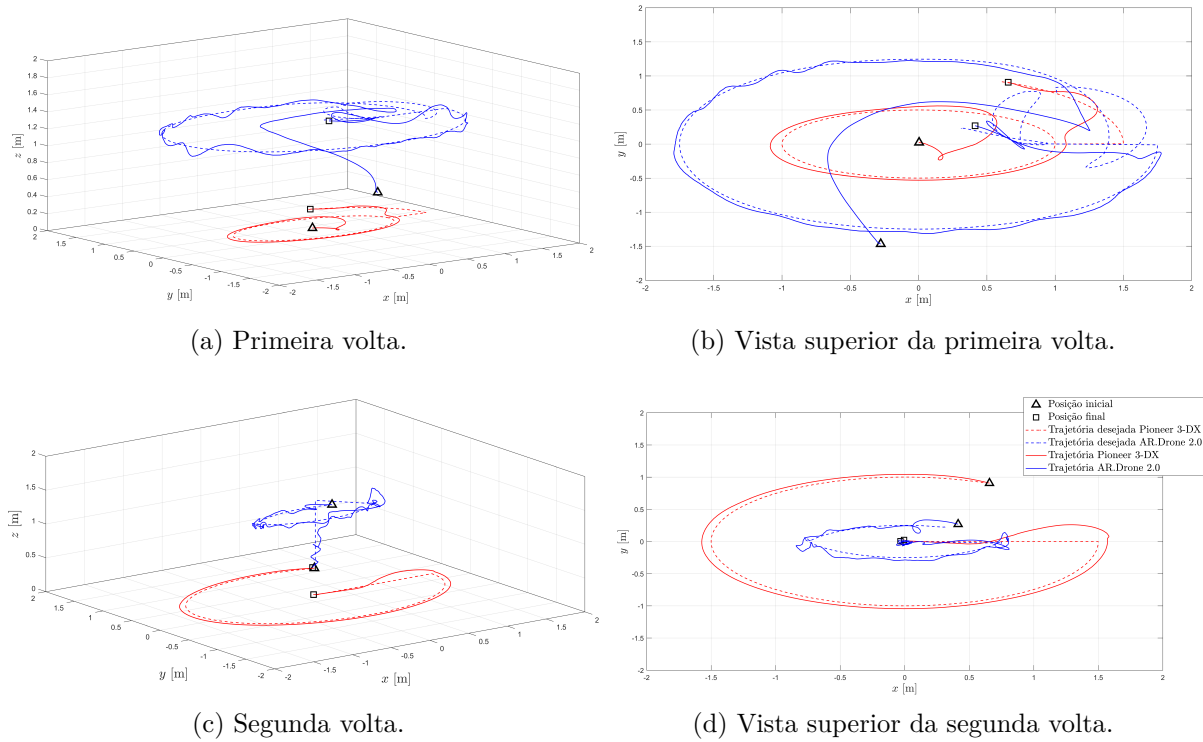


Figura 27 – Trajetória elíptica desejada e realizada separada em duas partes.

Tabela 9 – Parâmetros das variáveis de formação desejada para o experimento de trajetória elíptica com aterrissagem.

	x_f [m]	y_f [m]	z_f [m]	ρ_f [m]	α_f [graus]	β_f [graus]
1 ^a volta	$\cos(0, 15t)$	$0,5 \sin(0, 15t)$	0	1,5	$-90 + \psi_1$	60
2 ^a volta	$\cos(0, 15t)$	$0,5 \sin(0, 15t)$	0	1	$0 + \psi_1$	90
3 ^a volta	$\cos(0, 15t)$	$0,5 \sin(0, 15t)$	0	1	$-90 + \psi_1$	60
Pouso	0	0	0	$1,5 - (t - t_{total})/15$	$-90 + \psi_1$	60

4.2.3.3 Elipse com aterrissagem e decolagem

Também foram realizados experimentos com pouso e decolagem durante a trajetória elíptica¹⁰. Os parâmetros da formação podem ser vistos na Tabela 9. Após a primeira volta, β_f foi modificado para 90° e, então, o comando de aterrissagem foi acionado. A segunda volta foi executada pelo robô terrestre transportando o aéreo, e, ao final da segunda volta, foi dado o comando de decolagem. O VANT retornou à formação e continuou a trajetória na terceira volta. A Figura 32 contém detalhes desses momentos. Terminadas as três voltas, a formação foi enviada para o centro da elipse utilizando controle de posicionamento para finalizar o experimento com outra aterrissagem sobre a plataforma.

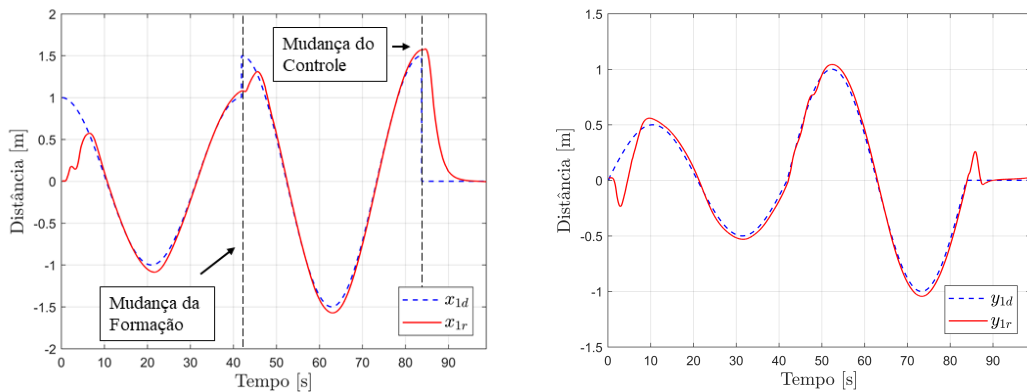
Para facilitar a visualização, o gráfico com a trajetória foi dividido em três, um para cada volta realizada, mostrados na Figura 33.

Durante a primeira volta (Figura 33a) os robôs acompanham a trajetória desejada corretamente. Na segunda volta (Figura 33b) o VANT aterrissou, e por essa razão não

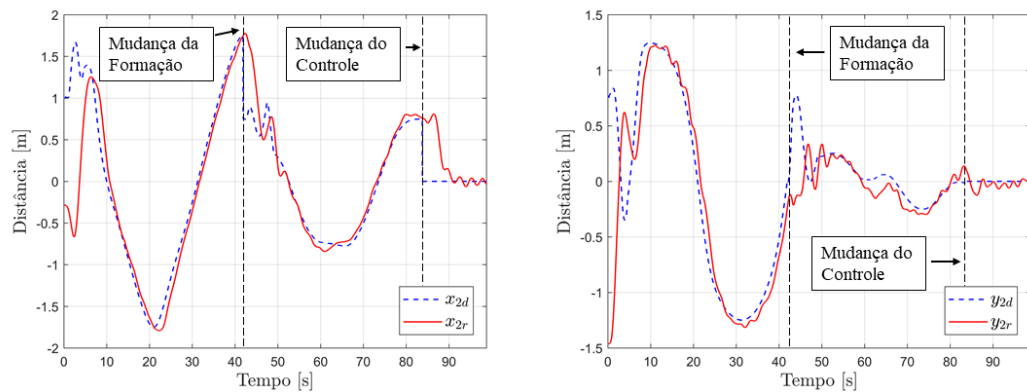
¹⁰ O vídeo do experimento pode ser acessado em <<https://www.youtube.com/watch?v=hi6tS2AMSzS>>.

manteve a altitude desejada. As posições em x e y entretanto coincidem com a do VTNT, de modo que o VANT atingiu as posições desejadas mesmo estando sobre a plataforma. Seguindo o experimento, o *drone* estabilizou após a decolagem e retornou à formação. Finalmente, após a terceira volta a formação foi enviada ao centro da elipse para a última aterrissagem e finalização do experimento.

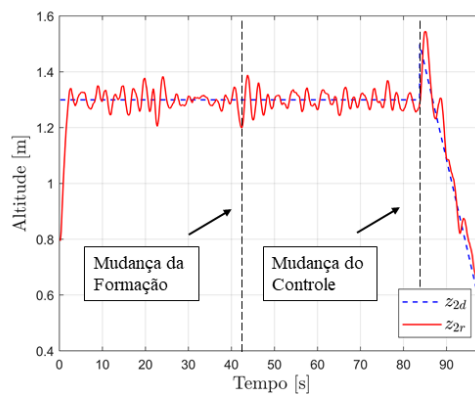
Uma oscilação principalmente do robô terrestre ocorreu nesse momento devido à mudança de controle de trajetória para posicionamento. Em controle de posicionamento



(a) Variação da posição em x do Pioneer 3-DX. (b) Variação da posição em y do Pioneer 3-DX.



(c) Variação da posição em x do AR.Drone 2.0. (d) Variação da posição em y do AR.Drone 2.0.



(e) Variação da posição em z do AR.Drone 2.0.

Figura 28 – Posições desejadas e realizadas pelos robôs na trajetória elíptica.

os erros de posição costumam ser maiores, resultando em maiores sinais de controle e logo os ganhos do controlador são menores. No controle de trajetória, por sua vez, os erros e consequentemente os sinais de controle são menores, necessitando normalmente de ganhos maiores para acompanhar a trajetória. Por isso, o chaveamento do controle sem a

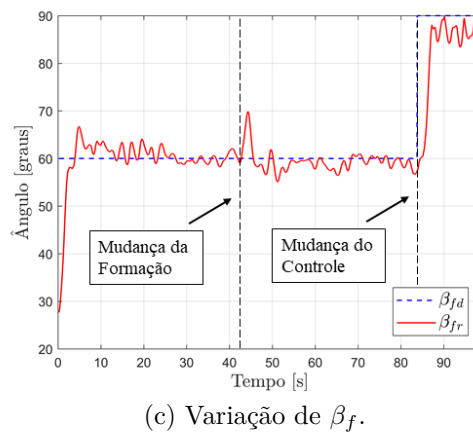
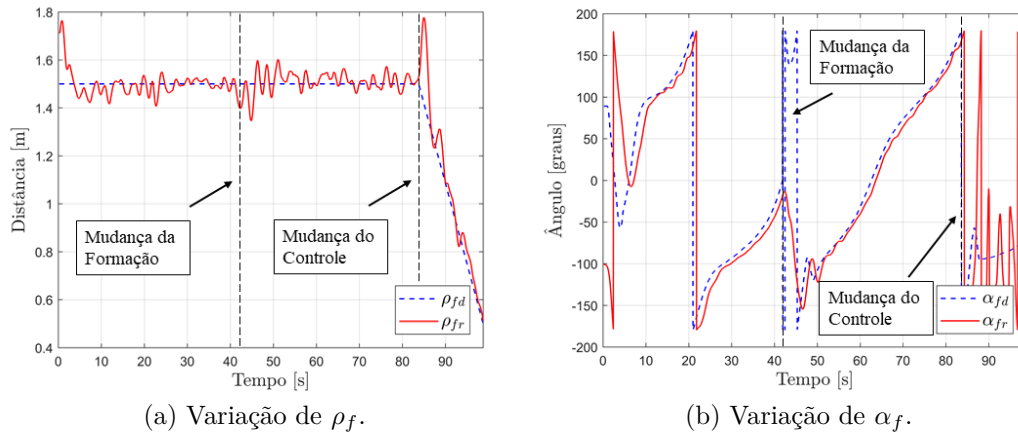


Figura 29 – Variáveis da formação ao longo da trajetória elíptica.

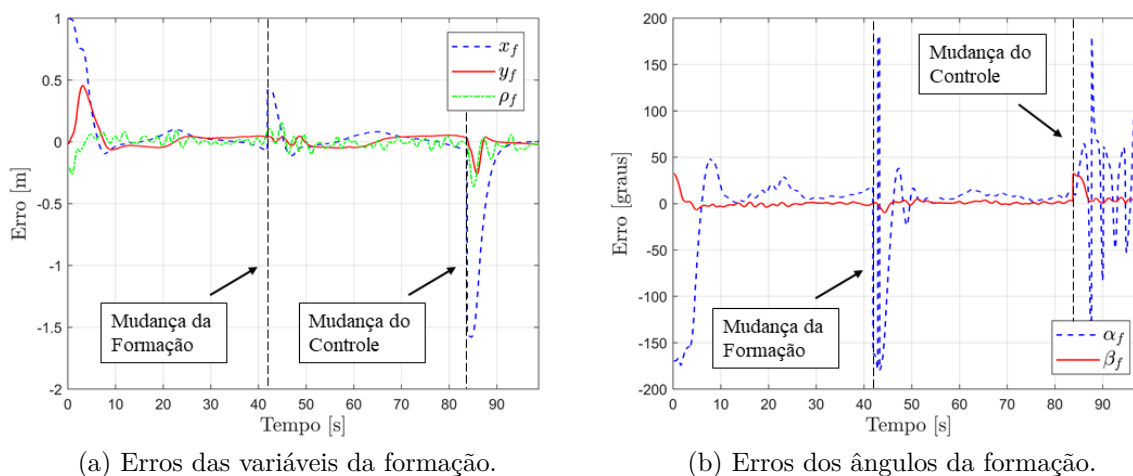


Figura 30 – Erros das variáveis de formação para trajetória elíptica.

adaptação dos ganhos pode gerar oscilações nos momentos iniciais como o observado nos experimentos.

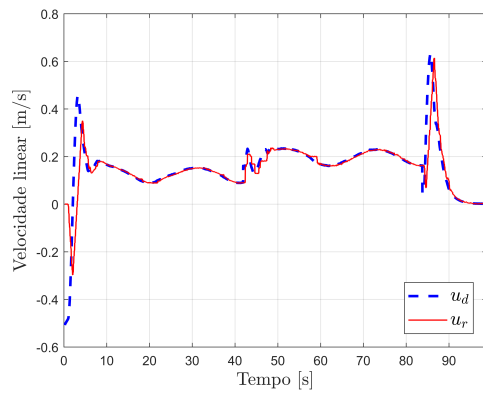
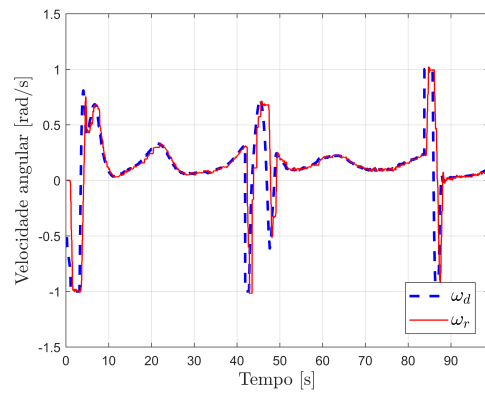
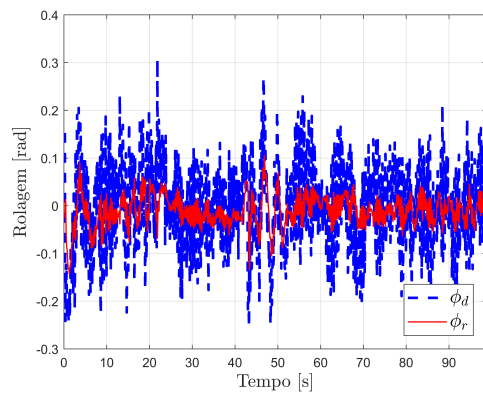
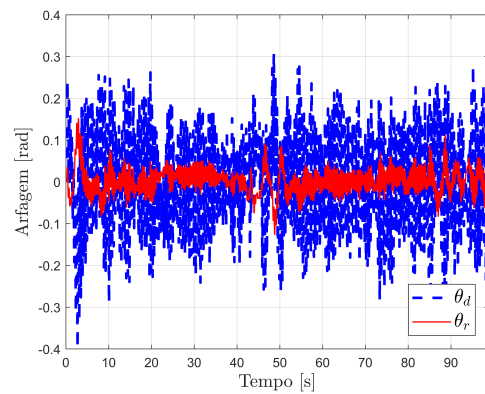
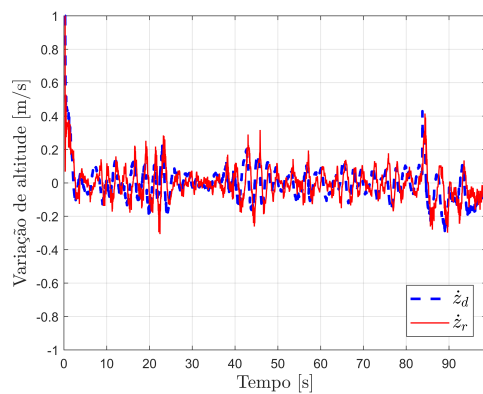
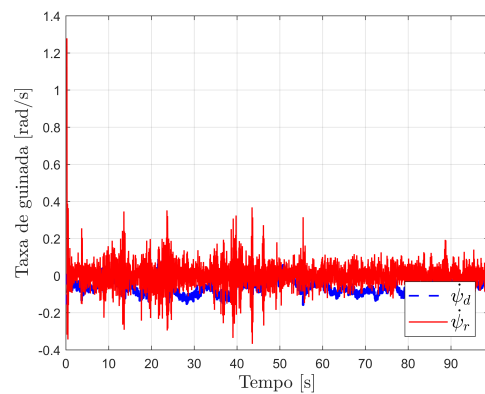
(a) Sinal de controle u do *Pioneer 3-DX*.(b) Sinal de controle ω do *Pioneer 3-DX*.(c) Sinal de controle ϕ do *AR.Drone 2.0*.(d) Sinal de controle θ do *AR.Drone 2.0*.(e) Sinal de controle \dot{z} do *AR.Drone 2.0*.(f) Sinal de controle $\dot{\psi}$ do *AR.Drone 2.0*.

Figura 31 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória elíptica.

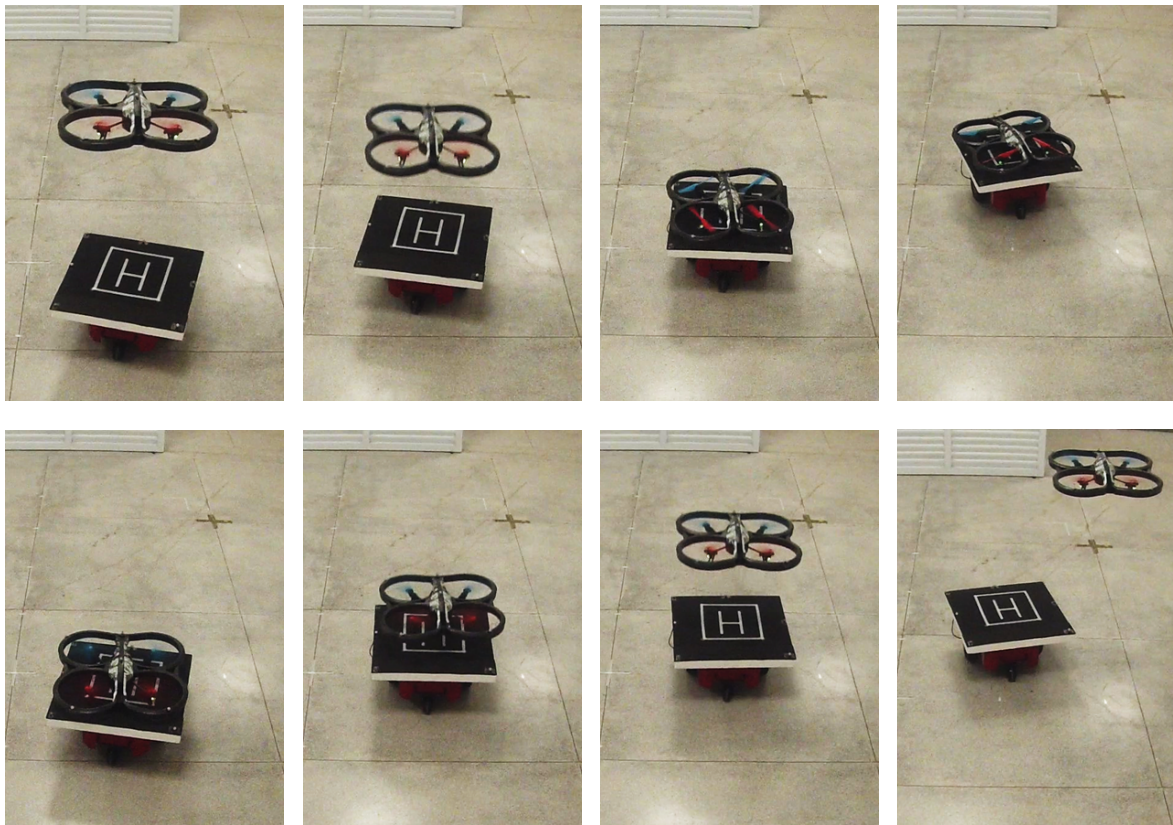


Figura 32 – Sequência de quadros respectivamente do momento da aterrissagem e decolagem no experimento de seguimento de trajetória elíptica.

Fonte: Autoria própria.

4.2.4 Simulação da formação homogênea aérea

A mesma estrutura de controle e equações utilizadas para a formação heterogênea pode ser aplicada para formações homogêneas. Como demonstração, foram realizadas simulações de seguimento de trajetória para dois *AR.Drone 2.0*. O percurso escolhido foi uma elipse inclinada. As variáveis de formação desejada são exibidas na Tabela 10.

A Figura 38a mostra os caminhos desejados e realizados pelos robôs. As variáveis da formação desejadas e realizadas são exibidas na Figura 38c. Todos os pontos desejados foram alcançados, comprovando a eficácia do controlador.

Um detalhe sobre esta formação específica é que deve a forma desejada ser considerada cuidadosamente, pois podem ocorrer distúrbios significativos entre os dois robôs. O vento gerado pelas hélices de um VANT pode desestabilizar o outro, dependendo de seus posicionamentos. Portanto um ângulo $\beta_f = 90^\circ$ por exemplo, na prática, pode atrapalhar a formação.

Porém, caso o modelo dos veículos inclua rejeição de distúrbio, em teoria é possível resolver essa questão até certo grau.

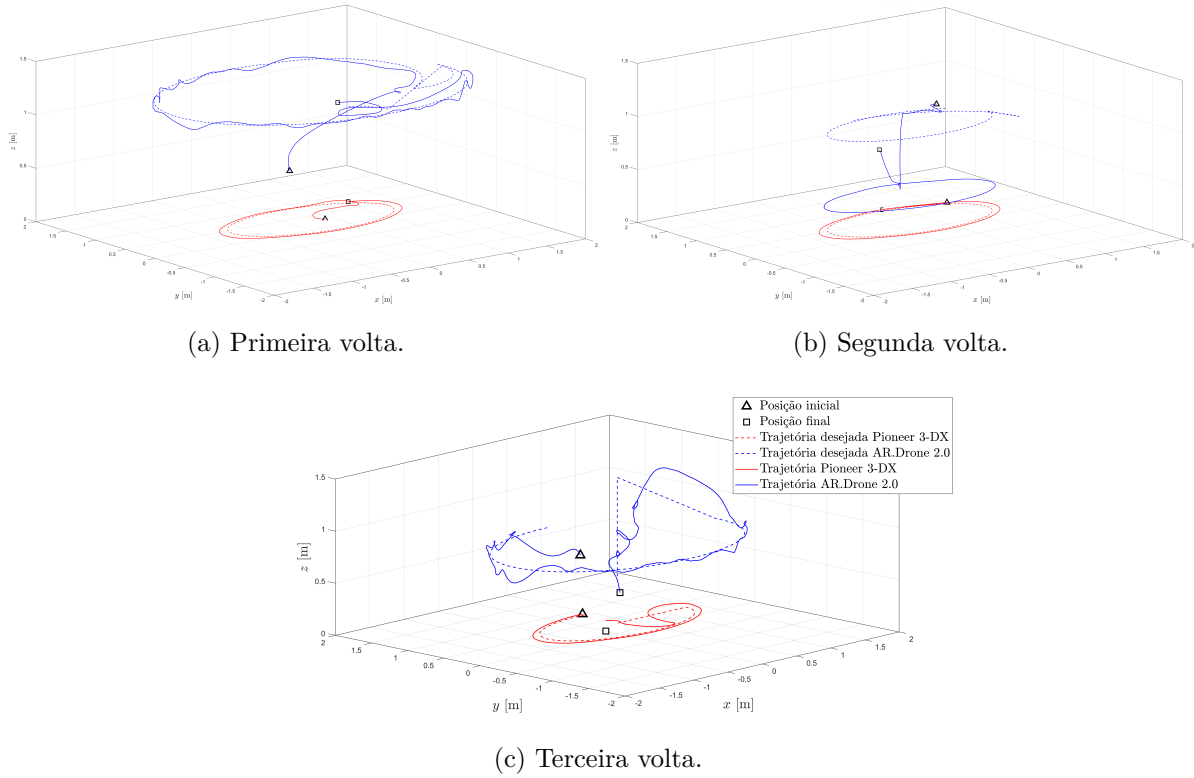
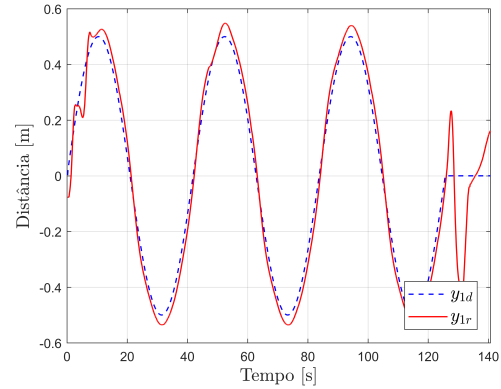
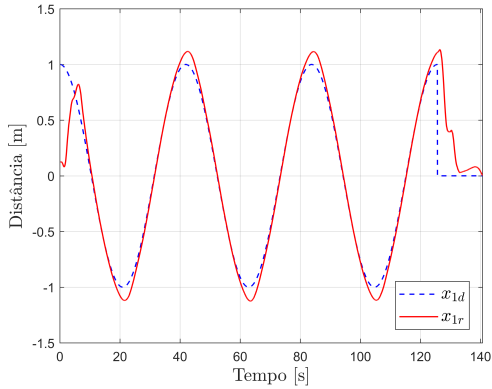


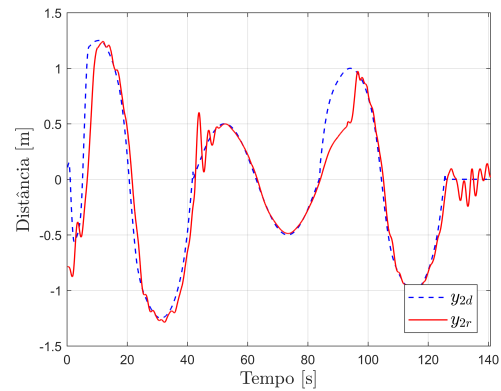
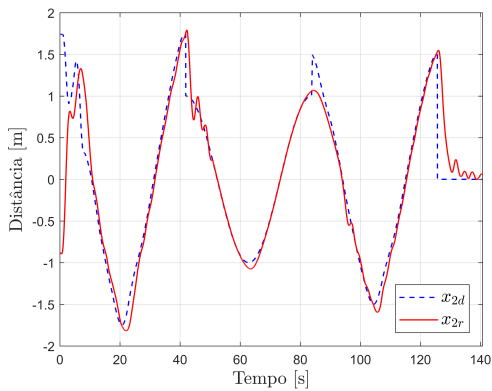
Figura 33 – Percurso desejado e realizado no experimento de trajetória elíptica com aterrisagem e decolagem durante o percurso.

Tabela 10 – Parâmetros das variáveis de formação desejada para a formação homogênea.

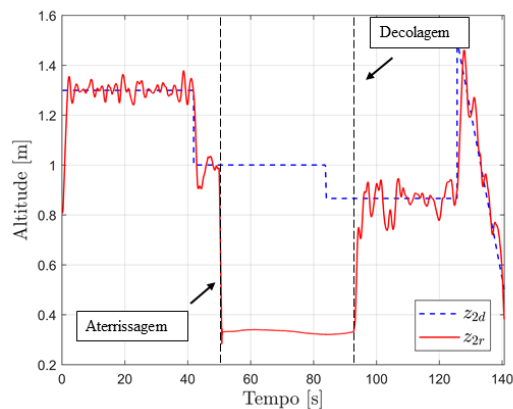
x_f [m]	y_f [m]	z_f [m]	ρ_f [m]	α_f [graus]	β_f [graus]
$3 \cos(0,05t)$	$2 \sin(0,05t)$	$3 + 2 \sin(0,05t)$	1,5	0,05t	45



(a) Variação da posição em x do *Pioneer 3-DX*. (b) Variação da posição em y do *Pioneer 3-DX*.



(c) Variação da posição em x do *AR.Drone 2.0*. (d) Variação da posição em y do *AR.Drone 2.0*.



(e) Variação da posição em z do *AR.Drone 2.0*.

Figura 34 – Posições desejadas e realizadas dos robôs ao longo do percurso elíptico com aterrissagem e decolagem durante o experimento.

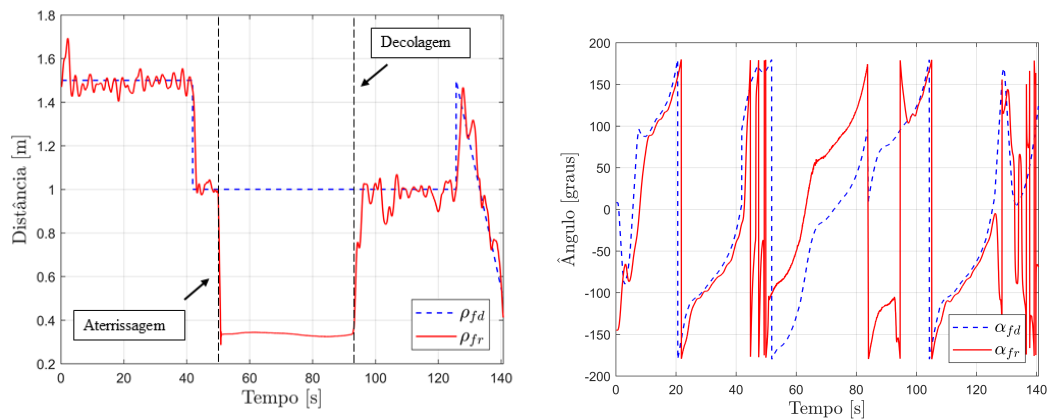
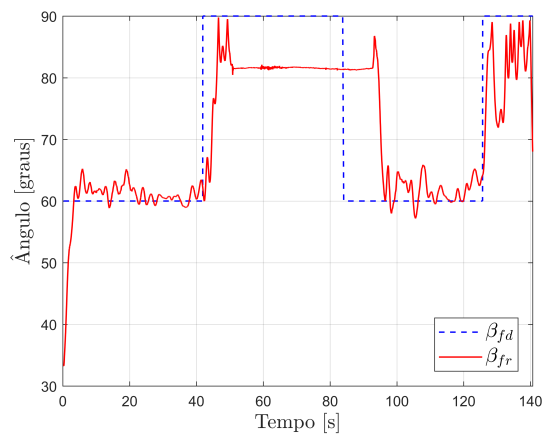
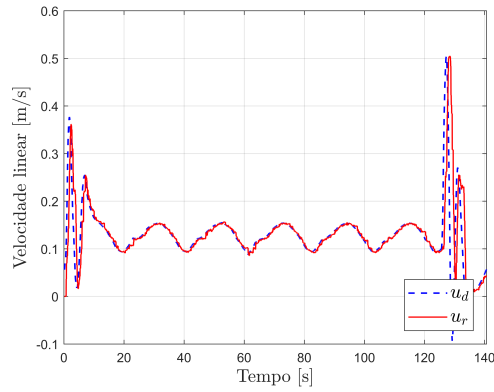
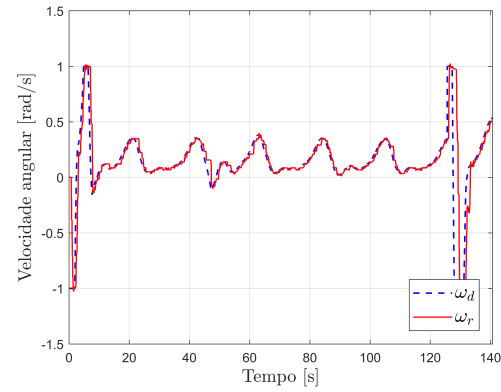
(a) Variação de ρ_f .(b) Variação de α_f .(c) Variação de β_f .

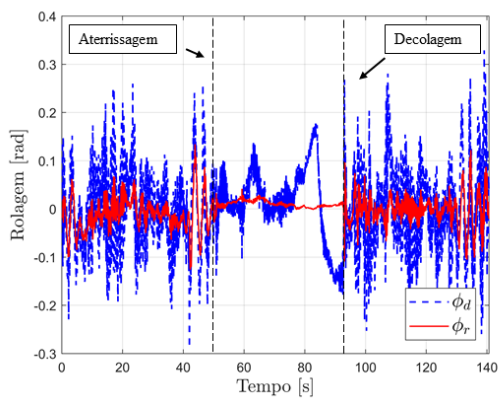
Figura 35 – Variáveis da formação ao longo do percurso.



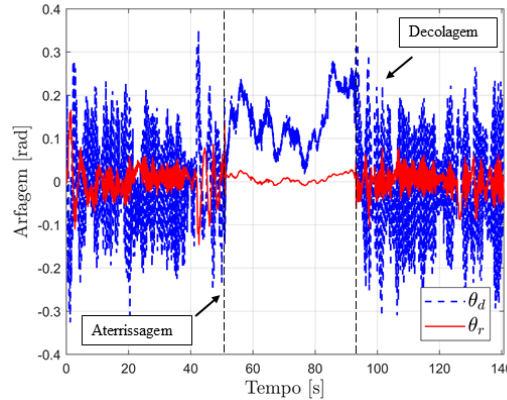
(a) Sinal de controle u desejado e realizado do Pioneer 3-DX.



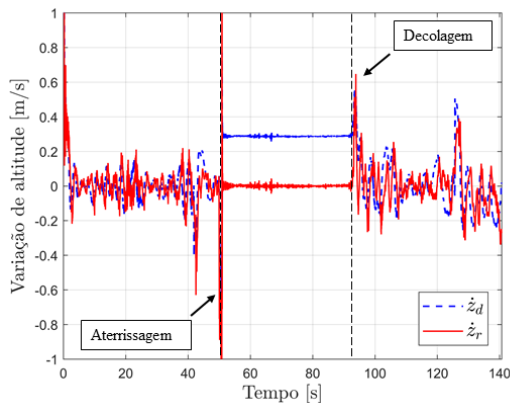
(b) Sinal de controle ω desejado e realizado do Pioneer 3-DX.



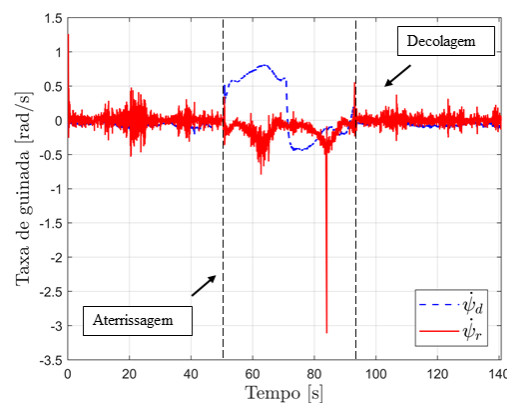
(c) Sinal de controle ϕ desejado e realizado do AR.Drone 2.0.



(d) Sinal de controle θ desejado e realizado do AR.Drone 2.0.

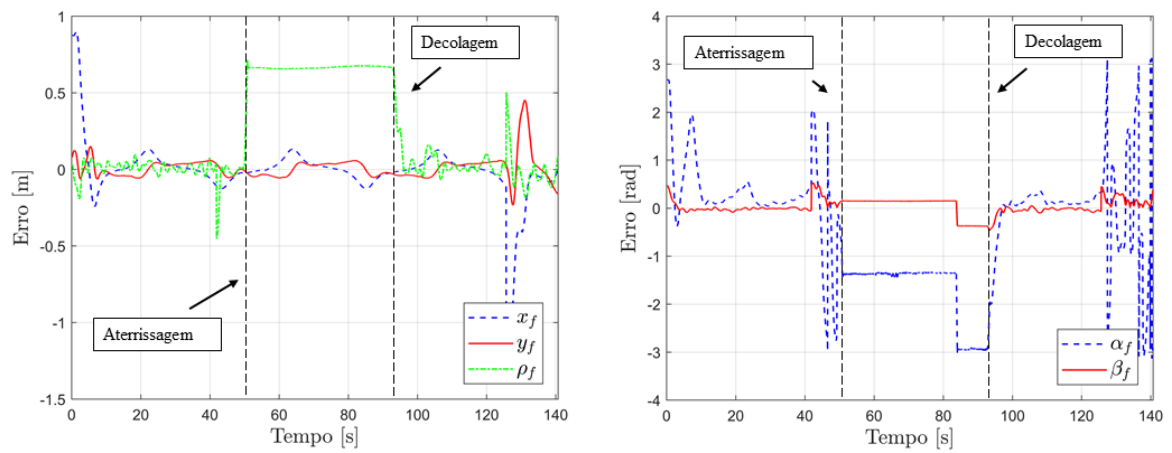


(e) Sinal de controle \dot{z} desejado e realizado do AR.Drone 2.0.



(f) Sinal de controle $\dot{\psi}$ desejado e realizado do AR.Drone 2.0.

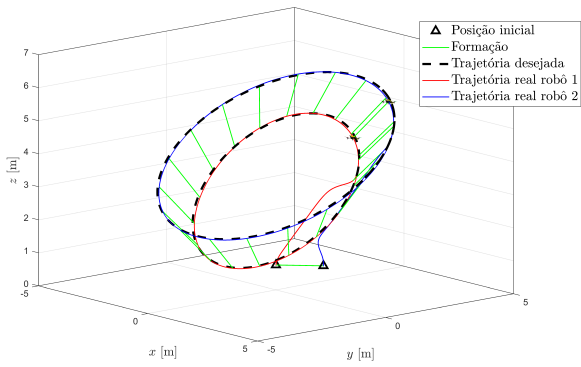
Figura 36 – Sinais de controle desejados e realizados dos robôs ao longo da trajetória.



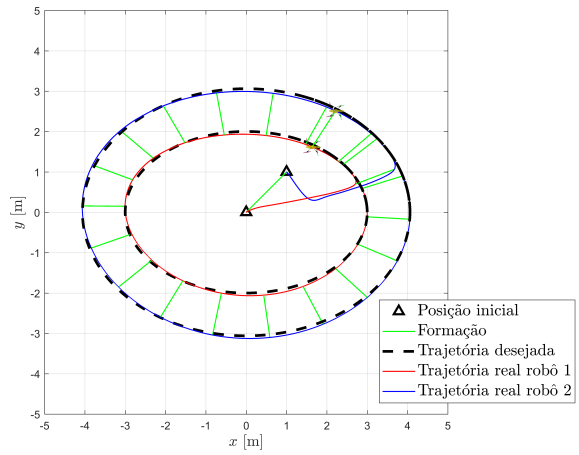
(a) Erros das variáveis da formação para o experimento de trajetória elíptica com aterrissagem durante o percurso.

(b) Erros dos ângulos da formação

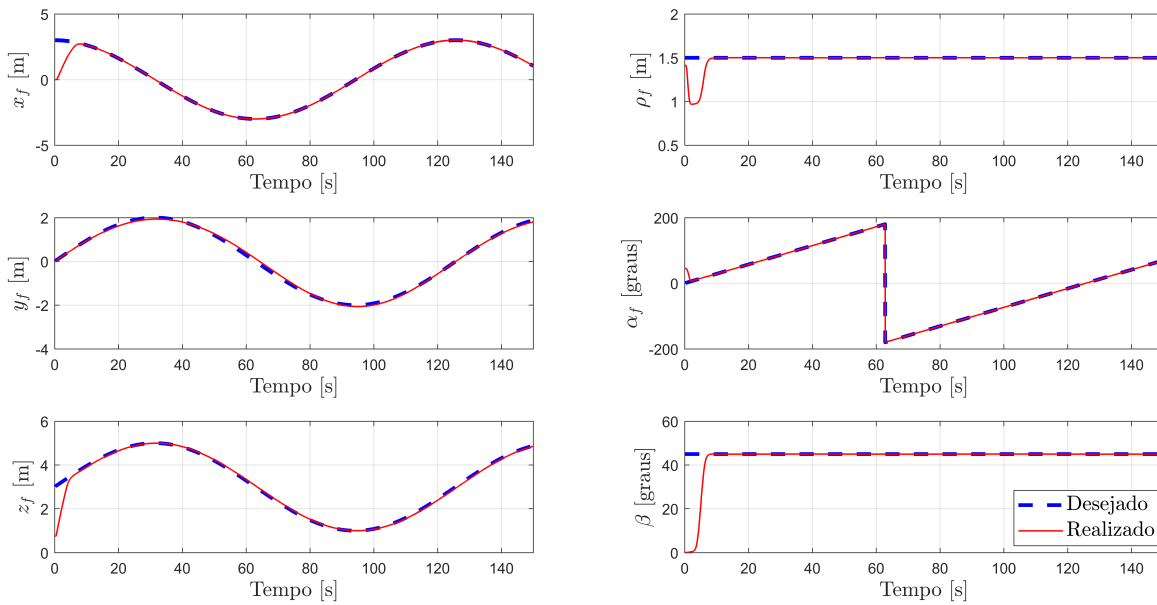
Figura 37 – Erros das variáveis de formação para trajetória elíptica.



(a) Percurso realizado pela formação homogênea em simulação.



(b) Vista superior do percurso.



(c) Evolução das variáveis de formação durante a simulação.

Figura 38 – Resultados da simulação da formação homogênea com dois VANT.

5 Conclusões

“A ciência nunca resolve um problema sem criar pelo menos outros dez.”

George Bernard Shaw

Foi possível através de simulações e experimentos validar o controlador proposto. Um ponto importante nesta pesquisa foi a possibilidade de avaliar o controle em uma situação perto da ideal, considerando que a grande dificuldade de navegação de robôs é a confiabilidade da informação de pose através dos sensores embarcados. A precisão do *OptiTrack* permitiu a eliminação desses erros na formação heterogênea. Esta precisão, além de auxiliar na navegação, é muito útil para identificação e validação de novos modelos, bem como aperfeiçoamento dos existentes.

A ideia do trabalho foi a simplificação do controle de formação para uma estrutura com apenas dois robôs, pensando na cooperação entre eles para transporte e entrega de cargas de modo autônomo. A grande vantagem deste método é não depender de um único robô líder para guiar a formação, os agentes navegam de forma coesa para manter a estrutura. No entanto, o modelo de controle multicamadas garante flexibilidade à formação, possibilitando alteração na forma para implementação de desvio de obstáculos por exemplo, o que aumentaria a capacidade de navegação do conjunto.

Além disso, o controlador é flexível a ponto de permitir, mesmo *on-line*, a troca do controle de posição para seguimento de trajetória e vice-versa. Isto seria útil, por exemplo, para enviar a formação a um ponto de repouso após finalizar sua tarefa, supondo que esta consista em seguimento de trajetória. Porém, é interessante ajustar ganhos proporcionais satisfatórios para as duas tarefas simultaneamente de modo a evitar oscilações no chaveamento. Outra estratégia é alterar também os ganhos juntamente com o tipo de controle.

O método de controle por estruturas virtuais utilizado possibilita o uso do mesmo sistema tanto para formações homogêneas como heterogêneas, sejam terrestres ou aéreas, sem necessidades de modificações extras. Para o transporte cooperativo de cargas, por exemplo, pode ser mais vantajoso uma formação homogênea, por facilitar a distribuição de peso entre os robôs, uma vez que possuem as mesmas características. Além de permitir a movimentação de cargas mais pesadas, ou com dimensões que dificultam o transporte com um único robô, esta configuração também resulta em economia de energia para os elementos da formação.

Ainda sobre a formação, é importante comentar que há restrições para as variáveis de formação que não foram discutidas neste trabalho. Por exemplo, a distância entre os robôs deve possuir sempre um valor mínimo, para evitar colisões entre eles. Valores máximos

também podem ser estabelecidos, para evitar colisões com estruturas do ambiente, ou perda de comunicação em casos extremos.

Vale ressaltar também que a formação 3D pode substituir a 2D, bastando fixar ou desconsiderar as variáveis de altura z_f e o ângulo vertical β_f . Mas, caso a aplicação seja com veículos terrestres em terreno plano, a formação 2D será mais vantajosa por possuir menos variáveis e equações, reduzindo o gasto com processamento desnecessário, assim como diminui o pacote de informações trocados entre eles. Isto é importante sabendo que a comunicação é um grande problema em controle de formação, e se considerarmos um possível aumento do número de agentes, otimização de recursos é essencial.

A parte experimental desta pesquisa evidenciou a complexidade de se trabalhar com sistemas reais. Muitos fatores devem ser considerados no desenvolvimento de um projeto, como protocolos de comunicação, escolha de *software* e linguagem de programação, capacidade física dos robôs, sensores disponíveis entre outros. Problemas de compatibilidade podem dificultar significativamente o projeto. Ou seja, para aplicações reais a parte técnica é tão importante quanto a teórica, e deve ser avaliada cuidadosamente.

De fato, a comunicação por exemplo é um fator crucial pra este método de controle. Pequenos atrasos ou perdas de pacotes podem afetar significativamente a estabilidade do sistema. Por isso o aumento de elementos de uma formação torna o controle centralizado inviável após certo número de agentes. Uma possibilidade é considerar esta formação como modular, de modo que cada robô acrescentado geraria uma nova formação em linha. Nesse caso um robô poderia fazer parte de duas formações simultaneamente, limitando a troca de informações mesmo com o aumento de agentes (obviamente, novas questões seriam geradas, como por exemplo: Um robô integrante de duas formações obedeceria a qual sinal de controle?).

O trabalho desenvolvido representa uma plataforma inicial para o problema de transporte de cargas, e portanto possui ainda limitações em relação a este objetivo final. Por esta razão é importante pensar nas próximas etapas.

O primeiro aspecto a se considerar é quanto ao ambiente de navegação. A formação terrestre atual depende dos sensores embarcados dos robôs e do alcance da rede *wireless* para troca de informação entre os agentes. E a formação heterogênea foi validada utilizando apenas sensores externos, estando restrita à navegação no volume limitado pela cobertura do *OptiTrack*. Mesmo que seja possível utilizar a estrutura de câmeras *outdoor*, o volume de captura será fixo (o que atenderia em certas situações, como em uma área delimitada dentro de um galpão, por exemplo).

Então mesmo aperfeiçoando o modelo dos robôs, ainda seria necessário outro método para tratar os erros cumulativos de odometria para ambos os sistemas, permitindo maior autonomia, visto que um sistema de captura de movimentos tem custo relativamente elevado, podendo ser inviável em muitos casos. Uma proposta mais barata é usar visão embarcada associada a sensor laser para auxílio da localização, através de técnicas de

SLAM (*Simultaneous Localization and Mapping*). A precisão provavelmente será menor, mas pode ser suficiente para a tarefa. Em ambientes externos favoráveis, seria possível ainda agregar informações provenientes de Sistemas de Navegação Global por Satélite (*Global Navigation Satellite System* (GNSS)). Outros métodos podem ser avaliados de acordo com o ambiente de navegação.

Outro fator de limitante do alcance para a formação heterogênea é o controle em um computador externo. Mas para solucionar basta que o robô terrestre possua um computador embarcado para ampliar as áreas autonomamente navegáveis. É importante que o computador tenha poder de processamento adequado às demandas da formação (soluções baseadas em imagens por exemplo usualmente têm elevado custo computacional).

Finalmente, levando em conta o que foi discutido, seguem algumas propostas para continuidade da pesquisa:

- Adaptação do controle para seguimento de caminho;
- Experimentos com formação aérea utilizando a plataforma desenvolvida;
- Comparação entre controle centralizado e descentralizado em relação à escalabilidade da formação.
- Aperfeiçoamento dos modelos dinâmicos dos robôs com auxílio do *OptiTrack*;
- Localização com apoio de visão embarcada (e ou laser) e fusão sensorial com dados de GPS para navegação *outdoor*;
- Acréscimo de algoritmos de desvio de obstáculo na camada de planejamento *On-line*;
- Transporte de carga cooperativo com formação homogênea e entrega de objetos com formação heterogênea.

Referências

- ABAUNZA, H. et al. Quaternion based control for circular uav trajectory tracking, following a ground vehicle: Real-time validation. *IFAC-PapersOnLine*, Elsevier, v. 50, n. 1, p. 11453–11458, 2017. Citado na página 21.
- ABBASI, Y.; MOOSAVIAN, S. A. A.; NOVINZADEH, A. B. Formation control of aerial robots using virtual structure and new fuzzy-based self-tuning synchronization. *Transactions of the Institute of Measurement and Control*, SAGE Publications Sage UK: London, England, v. 39, n. 12, p. 1906–1919, 2017. Citado na página 21.
- ADAMEK, T.; KITTS, C. A.; MAS, I. Gradient-based cluster space navigation for autonomous surface vessels. *IEEE/ASME Transactions on Mechatronics*, v. 20, n. 2, p. 506–518, April 2015. ISSN 1083-4435. Citado na página 19.
- AL-KAFF, A. et al. Survey of computer vision algorithms and applications for unmanned aerial vehicles. *Expert Systems with Applications*, Elsevier, 2017. Citado na página 20.
- ALONSO-MORA, J.; BAKER, S.; RUS, D. Multi-robot formation control and object transport in dynamic environments via constrained optimization. *The International Journal of Robotics Research*, SAGE Publications Sage UK: London, England, v. 36, n. 9, p. 1000–1021, 2017. Citado na página 22.
- ANTONELLI, G.; ARRICHIELLO, F.; CHIAVERINI, S. The nsb control: a behavior-based approach for multi-robot systems. *Paladyn*, v. 1, n. 1, p. 48–56, March 2010. ISSN 2080-9778. Citado na página 19.
- ARBANAS, B. et al. Aerial-ground robotic system for autonomous delivery tasks. In: IEEE. *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. [S.l.], 2016. p. 5463–5468. Citado na página 19.
- BALCH, T.; ARKIN, R. C. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, v. 14, n. 6, p. 926–939, December 1998. ISSN 1042-296X. Citado na página 19.
- BOEL, R. K.; MARINICA, N. E.; SARLETTE, A. Leader-follower cooperative control paradigm, with applications to urban traffic coordination control. In: *2015 European Control Conference (ECC)*. [S.l.: s.n.], 2015. p. 2208–2215. Citado na página 19.
- BRANDÃO, A. S. et al. The multilayer control scheme: A strategy to guide n -robots formations with obstacle avoidance. *Journal of Control, Automation and Electrical Systems*, v. 26, n. 3, p. 201–214, June 2015. ISSN 2195-3899. Citado 3 vezes nas páginas 19, 30 e 33.
- BRANDÃO, A. S.; SARCINELLI-FILHO, M. On the guidance of multiple uav using a centralized formation control scheme and delaunay triangulation. *Journal of Intelligent & Robotic Systems*, Springer, v. 84, n. 1-4, p. 397–413, 2016. Citado 2 vezes nas páginas 19 e 33.

- BRANDÃO, A. S.; SARCINELLI-FILHO, M.; CARELLI, R. High-level underactuated nonlinear control for rotorcraft machines. In: IEEE. *Mechatronics (ICM), 2013 IEEE International Conference on*. [S.l.], 2013. p. 279–285. Citado na página 28.
- BRANDÃO, A. S. et al. Decentralized control of leader-follower formations of mobile robots with obstacle avoidance. In: *2009 IEEE International Conference on Mechatronics*. Malaga, Spain: [s.n.], 2009. p. 1–6. Citado na página 19.
- CAI, Y.; KOUSUKE, S. Geometric relation matching based object identification for uav and ugv cooperation. In: IEEE. *Technologies and Applications of Artificial Intelligence (TAAI), 2015 Conference on*. [S.l.], 2015. p. 478–483. Citado na página 19.
- CHRISTIE, G. et al. Radiation search operations using scene understanding with autonomous uav and ugv. *Journal of Field Robotics*, Wiley Online Library, v. 34, n. 8, p. 1450–1468, 2017. Citado 2 vezes nas páginas 19 e 20.
- GUILLET, A. et al. Formation control of agricultural mobile robots: A bidirectional weighted constraints approach. *Journal of Field Robotics*, Wiley Online Library, v. 34, n. 7, p. 1260–1274, 2017. Citado na página 21.
- HOOD, S. et al. Bird’s eye view: Cooperative exploration by ugv and uav. In: IEEE. *Unmanned Aircraft Systems (ICUAS), 2017 International Conference on*. [S.l.], 2017. p. 247–255. Citado na página 19.
- JASIM, W.; GU, D. Robust team formation control for quadrotors. *IEEE Transactions on Control Systems Technology*, IEEE, v. 26, n. 4, p. 1516–1523, 2018. Citado na página 22.
- KHALEGHI, A. M. et al. Analysis of uav/ugv control strategies in a dddams-based surveillance system. In: INSTITUTE OF INDUSTRIAL AND SYSTEMS ENGINEERS (IISE). *IIE Annual Conference. Proceedings*. [S.l.], 2015. p. 2283. Citado na página 19.
- KITTS, C. A.; MAS, I. Cluster space specification and control of mobile multirobot systems. *IEEE/ASME Transactions on Mechatronics*, v. 14, n. 2, p. 207–218, April 2009. ISSN 1083-4435. Citado 2 vezes nas páginas 19 e 30.
- LI, H. et al. Multivehicle cooperative local mapping: A methodology based on occupancy grid map merging. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 15, n. 5, p. 2089–2100, 2014. Citado na página 18.
- LI, Z.; DUAN, Z.; HUANG, L. Leader-follower consensus of multi-agent systems. In: *2009 American Control Conference*. [S.l.: s.n.], 2009. p. 3256–3261. ISSN 0743-1619. Citado na página 19.
- LIANG, X. et al. Leader-following formation tracking control of mobile robots without direct position measurements. *IEEE Transactions on Automatic Control*, IEEE, v. 61, n. 12, p. 4131–4137, 2016. Citado na página 19.
- LIANG, X. et al. Formation control of nonholonomic mobile robots without position and velocity measurements. *IEEE Transactions on Robotics*, IEEE, v. 34, n. 2, p. 434–446, 2018. Citado na página 21.
- LUDVIGSEN, M. et al. Network of heterogeneous autonomous vehicles for marine research and management. In: IEEE. *OCEANS 2016 MTS/IEEE Monterey*. [S.l.], 2016. p. 1–7. Citado na página 20.

- MARTINS, F. N. *Modelagem e compensação da dinâmica de robôs móveis e sua aplicação em controle de formação*. Tese (Doutorado) — PhD thesis, Universidade Federal do Espírito Santo, 2009. Citado 2 vezes nas páginas 27 e 39.
- MARTINS, F. N.; SARCINELLI-FILHO, M.; CARELLI, R. A velocity-based dynamic model and its properties for differential drive mobile robots. *Journal of Intelligent & Robotic Systems*, Springer, v. 85, n. 2, p. 277–292, 2017. Citado na página 26.
- MASONE, C.; BÜLTHOFF, H. H.; STEGAGNO, P. Cooperative transportation of a payload using quadrotors: A reconfigurable cable-driven parallel robot. In: IEEE. *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. [S.l.], 2016. p. 1623–1630. Citado na página 18.
- NGUYEN, T.; GARONE, E. Control of a uav and a ugv cooperating to manipulate an object. In: IEEE. *American Control Conference (ACC), 2016*. [S.l.], 2016. p. 1347–1352. Citado na página 19.
- OYEKAN, J. Bio-inspired vision-based leader-follower formation flying in the presence of delays. *Robotics*, Multidisciplinary Digital Publishing Institute, v. 5, n. 3, p. 18, 2016. Citado na página 21.
- PENG, L. et al. Decentralized multi-robot formation control with communication delay and asynchronous clock. *Journal of Intelligent & Robotic Systems*, Springer, v. 89, n. 3-4, p. 465–484, 2018. Citado na página 22.
- PISKORSKI, S. et al. *AR.Drone Developer Guide, Parrot SDK 2.0*. [S.l.], May 2012. Citado na página 27.
- QIANG, L. et al. Formation control of multi robot based on uwb distance measurement. In: IEEE. *2018 Chinese Control And Decision Conference (CCDC)*. [S.l.], 2018. p. 2404–2408. Citado na página 22.
- ROSALES, C. et al. 3d formation control of autonomous vehicles based on null-space. *Journal of Intelligent & Robotic Systems*, Springer, v. 84, n. 1-4, p. 453–467, 2016. Citado 2 vezes nas páginas 19 e 21.
- SANTANA, L. V.; BRANDAO, A. S.; SARCINELLI-FILHO, M. Outdoor waypoint navigation with the ar. drone quadrotor. In: IEEE. *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*. [S.l.], 2015. p. 303–311. Citado na página 27.
- SANTANA, L. V.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Heterogeneous leader-follower formation based on kinematic models. In: IEEE. *Unmanned Aircraft Systems (ICUAS), 2016 International Conference on*. [S.l.], 2016. p. 342–346. Citado na página 25.
- SANTOS, M. C. P. *Controle em Ambientes Interiores de Veículos Aéreos Não Tripulados*. Tese (Doutorado) — Universidade Federal do Espírito Santo, 2017. Citado na página 20.
- SAVIC, S. P. et al. Clinical application of a portable motion capture system: A methodology. In: IEEE. *2018 7th Mediterranean Conference on Embedded Computing (MECO)*. [S.l.], 2018. p. 1–4. Citado na página 76.

- SCHANG, A. C. et al. Quadrotor performance model verification and validation. In: *2018 AIAA Modeling and Simulation Technologies Conference*. [S.l.: s.n.], 2018. p. 1922. Citado na página 20.
- SCHNEIDER, F. E.; WILDERMUTH, D. Using robots for firefighters and first responders: Scenario specification and exemplary system description. In: IEEE. *Carpathian Control Conference (ICCC), 2017 18th International*. [S.l.], 2017. p. 216–221. Citado na página 17.
- SICILIANO, B.; KHATIB, O. *Springer Handbook of Robotics*. [S.l.]: Springer International Publishing, 2016. (Springer Handbooks). ISBN 9783319325521. Citado 2 vezes nas páginas 17 e 20.
- SIEGWART, R. et al. *Introduction to autonomous mobile robots*. [S.l.]: MIT press, 2011. Citado na página 20.
- SURMANN, H.; BERNINGER, N.; WORST, R. 3d mapping for multi hybrid robot cooperation. In: IEEE. *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. [S.l.], 2017. p. 626–633. Citado na página 20.
- TOKEKAR, P. et al. Sensor planning for a symbiotic uav and ugv system for precision agriculture. *IEEE Transactions on Robotics*, IEEE, v. 32, n. 6, p. 1498–1511, 2016. Citado na página 19.
- TZAFESTAS, S. G. 11 - mobile robot path, motion, and task planning. In: TZAFESTAS, S. G. (Ed.). *Introduction to Mobile Robot Control*. Oxford: Elsevier, 2014. p. 429 – 478. ISBN 978-0-12-417049-0. Citado na página 17.
- URCOLA, P. et al. Cooperative minimum expected length planning for robot formations in stochastic maps. *Robotics and Autonomous Systems*, Elsevier, v. 87, p. 38–50, 2017. Citado na página 22.
- WAN, S.; LU, J.; FAN, P. Semi-centralized control for multi robot formation. In: IEEE. *Robotics and Automation Engineering (ICRAE), 2017 2nd International Conference on*. [S.l.], 2017. p. 31–36. Citado na página 22.
- WANG, H. et al. Adaptive vision-based leader–follower formation control of mobile robots. *IEEE Transactions on Industrial Electronics*, IEEE, v. 64, n. 4, p. 2893–2902, 2017. Citado na página 21.
- YAMCHI, M. H.; ESFANJANI, R. M. Distributed predictive formation control of networked mobile robots subject to communication delay. *Robotics and Autonomous Systems*, Elsevier, v. 91, p. 194–207, 2017. Citado 2 vezes nas páginas 19 e 22.
- ZHANG, J. et al. Mobile robot for retail inventory using rfid. In: IEEE. *Industrial Technology (ICIT), 2016 IEEE International Conference on*. [S.l.], 2016. p. 101–106. Citado na página 19.
- ZHU, J.; ZHU, J.; XU, C. A testbed for aerial robots formation flight. In: *Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA)*. [S.l.: s.n.], 2016. p. 1183–1188. Citado na página 20.

APÊNDICE A – *OptiTrack*

OptiTrack é um sistema de captura de movimentos pertencente à empresa NaturalPoint Inc, utilizado mundialmente por grandes grupos como Nasa, Google, Nike, Activision e Disney. Dentre suas aplicações podem ser citadas animações, realidade virtual, estudo do movimento e robótica. Ele funciona através de câmeras posicionadas em diversos ângulos que emitem sinais infravermelhos e capturam imagens do ambiente, e é controlado pelo programa Motive. Um exemplo de montagem para quatro câmeras pode ser visualizado na Figura 39.

Para capturar os movimentos dos corpos de interesse, marcadores compostos de materiais reflexivos ao infravermelho são acoplados ao objeto em questão, e mediante o processamento das imagens capturadas em diferentes ângulos, são obtidas as coordenadas tridimensionais de cada marcador em relação ao eixo de referência global, que é determinado durante a calibração.

A calibração das câmeras é feita por duas ferramentas, mostradas na Figura 40. Primeiramente, o bastão da Figura 40a deve ser movimentado por todo o ambiente. Sempre que os marcadores entram no campo de visão de pelo menos duas câmeras simultaneamente, os pontos são salvos pelo programa. Quando um número suficiente de pontos for capturado (segundo o fabricante, até 10000 pontos para cada câmera, mais que isso começa a atrapalhar os cálculos), o processo deve ser finalizado e o programa calculará o volume de captura.

O procedimento anterior define a distância relativa entre as câmeras, mas não consegue posicioná-las em relação ao eixo vertical. Para gerar a referência de altitude em relação ao piso, o esquadro mostrado na Figura 40b deve ser utilizado. Ao posicioná-lo no chão, o marcador na quina do esquadro será a origem do eixo de coordenadas global, o lado

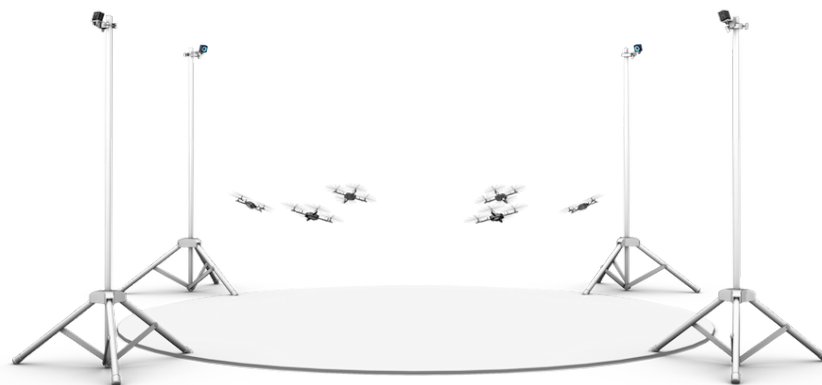


Figura 39 – Esquema de ambiente para experimentos com o *OptiTrack*.

Fonte: Disponível em <<https://optitrack.com/systems/#robotics/prime-13/4>>. Acesso em 20 nov. 2018.

de menor comprimento é o eixo X , e o lado maior o eixo Z ¹. Com os três marcadores do esquadro, o referencial de altitude zero é definido, e as posições das câmeras são recalculadas. O sistema estará então calibrado, e o processo não precisa ser repetido enquanto as câmeras não forem deslocadas de suas posições.

O *software* Motive permite a criação de esqueletos e corpos rígidos. Esqueletos são utilizados principalmente para captura de movimentos do corpo humano. Neste caso, devido à maior complexidade de processamento, os marcadores devem ser posicionados em regiões predefinidas do corpo para simplificar os cálculos. Além dos casos citados no início da seção, há muita aplicação na área da saúde. Em (SAVIC et al., 2018) por exemplo, o *OptiTrack* foi utilizado para auxiliar na modelagem dos movimentos do joelho, permitindo comparações entre ligamentos saudáveis e com problemas.

Já para o corpo rígido, são necessários ao menos três marcadores, e não há uma distância predefinida entre eles. Segundo informações do fabricante, o número de marcadores recomendado para melhor captura é de cinco a doze. Para criar um corpo rígido, primeiramente o programa deve estar operando no modo *on-line*. Neste estado, é exibido na tela do Motive o ambiente de captura em tempo real. Logo, é preciso selecionar os pontos referentes aos marcadores desejados, e defini-los como corpo rígido. O programa automaticamente calcula seu centroide, cuja orientação coincide com o eixo de coordenadas global, e o assinala como centro de referência. A referência pode ser reposicionada livremente, o que evita que o usuário tenha que fazer cálculos posteriores ou colocar os marcadores em locais específicos para obter a coordenada do ponto de controle do robô.

Uma vez que o sistema considera que não haverá alteração das distâncias entre os marcadores (por ser um corpo rígido), caso alguns marcadores saiam do campo de visão das câmeras, ainda é possível calcular a posição atual do robô desde que ao menos três

¹ O *software* opera com o eixo Y apontando para cima. No entanto, é possível modificar esta orientação para o padrão com eixo Z para cima nas configurações de envio de dados para a rede.

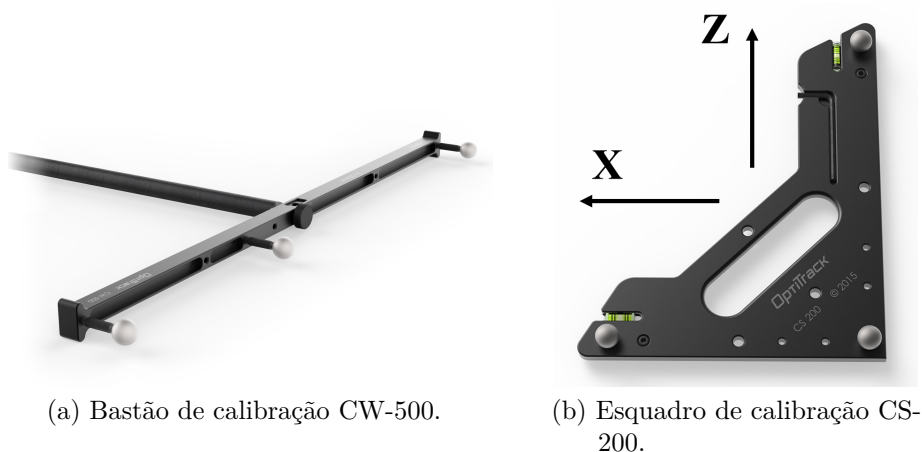


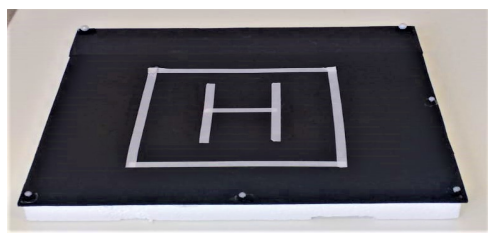
Figura 40 – Ferramentas para calibração das câmeras do *OptiTrack*.
Fonte: Disponível em <<https://optitrack.com/products/tools/>>. Acesso em 20 nov. 2018.

sejam rastreados. Por este motivo é importante colocar mais do que três marcadores em cada corpo rígido. Além disso, é indicado posicionar os marcadores de maneira assimétrica, montando um padrão único para cada robô. Deste modo evita-se problemas durante a identificação de cada objeto pelo programa.

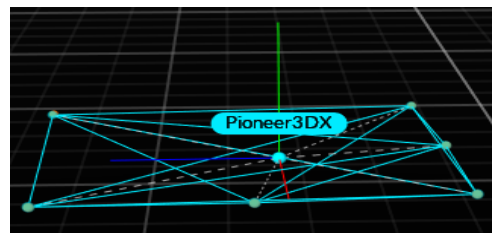
A detecção dos marcadores é diretamente influenciada pela iluminação, e portanto há configurações personalizáveis para adaptar a captura às condições do ambiente. É possível configurar itens como intensidade do infravermelho, exposição das câmeras, FPS, *Threshold* da segmentação da imagem, grau de circularidade dos marcadores entre outras opções, que interferem significativamente nos resultados. Os experimentos foram realizados em diferentes horários, e o ambiente estava exposto à iluminação externa. Conseqüentemente, as configurações do programa tiveram que ser ajustadas para cada caso.

Nos experimentos foram utilizadas quatro câmeras do modelo Prime 13, posicionadas em média a 3m de altura formando algo próximo a um prisma retangular, similar à Figura 39. Elas podem capturar imagens na frequência de até 240 FPS (*frames* por segundo). Quanto aos robôs, foram anexados sete marcadores na carcaça de proteção do *ArDrone 2.0* e seis na plataforma colocada sobre o *Pioneer 3-DX*, como mostrado na Figura 41. A figura mostra cada objeto com seu respectivo corpo rígido criado no Motive 2.0.

Para execução do experimento, o Motive 2.0 e o MatlabR2017a devem operar simultaneamente. No modo *on-line*, o Motive 2.0 envia automaticamente para rede uma *string* contendo dados que podem ser configuradas pelo usuário. As informações padrão são *frame* atual, as identificações, posições, orientações e matrizes de rotação dos corpos rígidos. É possível também enviar dados de posição de cada marcador se necessário. Estes dados são



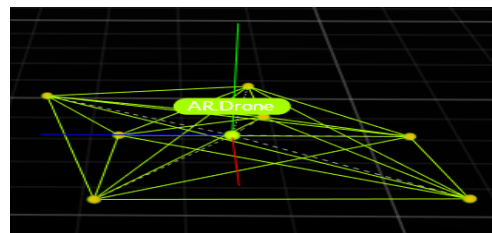
(a) Plataforma de pouso com seis marcadores.



(b) Corpo rígido criado para a plataforma de pouso.



(c) *ArDrone 2.0* com sete marcadores.



(d) Corpo rígido criado para o *ArDrone 2.0*.

Figura 41 – Robôs com seus marcadores e os respectivos corpos rígidos criados no Motive 2.0.

capturados pelo MatlabR2017a, que utiliza a informação para obter a pose atual de cada robô. A partir da pose, são obtidas as velocidades por derivação.

O sistema *OptiTrack* foi utilizado apenas para os experimentos com a formação heterogênea, porque, quando o equipamento ficou disponível, os experimentos com a formação terrestre já haviam sido finalizados. E como a odometria do *Pioneer 3-DX* produziu resultados satisfatórios, estes experimentos foram mantidos. Outro fator foi o espaço reduzido do ambiente de captura com as câmeras, que limitaria muito a navegação da formação com dois veículos terrestres.