

Vinícius Pacheco Bacheti

**Seguimento de Caminho por uma Formação Robô Terrestre - Robô  
Aéreo**

Vitória, Espírito Santo  
2021



Vinícius Pacheco Bacheti

## **Seguimento de Caminho por uma Formação Robô Terrestre - Robô Aéreo**

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  
2021

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

---

B122s Bacheti, Vinícius Pacheco, 1992-  
Seguimento de caminho por uma formação robô terrestre - robô aéreo / Vinícius Pacheco Bacheti. - 2021.  
78 f. : il.

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

1. Engenharia elétrica. 2. Robótica. 3. Robôs móveis. I.  
Sarcinelli Filho, Mário. II. Brandão, Alexandre Santos. III.  
Universidade Federal do Espírito Santo. Centro Tecnológico. IV.  
Título.

CDU: 621.3

---

VINÍCIUS PACHECO BACHETI

SEGUIMENTO DE CAMINHO POR UMA FORMAÇÃO ROBÔ  
TERRESTRE - ROBÔ AÉREO

Dissertação submetida ao programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para a obtenção do Grau de Mestre em Engenharia Elétrica.

Aprovada em 05 de março de 2021.

COMISSÃO EXAMINADORA



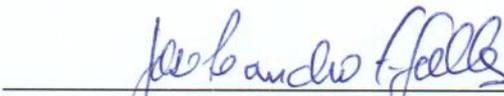
Prof. Dr. Mário Sarcinelli Filho  
Universidade Federal do Espírito Santo  
Orientador



Prof. Dr. Alexandre Santos Brandão  
Universidade Federal de Viçosa  
Coorientador



Profa. Dra. Eliete Maria de Oliveira Caldeira  
Universidade Federal do Espírito Santo



Prof. Dr. José Leandro Félix Salles  
Universidade Federal do Espírito Santo



Prof. Dr. Lucas Vago Santana  
Instituto Federal de Educação, Ciência e Tecnologia do  
Espírito Santo - Campus Linhares



*A todos aqueles que amo.*



# Agradecimentos

Agradeço primeiramente a minha família que me deu muito além do suporte necessário para que eu pudesse concluir com sucesso essa jornada. Aos meus pais, Gilmar e Luciane, fica a minha mais profunda gratidão por tudo o que me ensinaram e os valores que me passaram. Vou levar isso comigo por toda a minha vida. Aos meus irmãos, Pedro e Gilmara, agradeço pelo companheirismo e carinho, assim como a preocupação de saber se o "irmãozinho" deles estava bem. Não poderia deixar de citar as minhas sobrinhas, Elisa e Isadora, que têm trazido tanta alegria e leveza para a minha vida. Tio Bini ama vocês. Por último, agradeço de coração a minha namorada, Jaquelina, que foi quem mais viu meus sucessos, aguentou a rabugice nas minhas falhas e foi a melhor companheira que poderia pedir nestes anos.

Ao meu orientador, Professor Sarcinelli, espero ter correspondido à confiança que foi depositada em mim e agradeço sua orientação e ajuda durante toda esta jornada. A pesquisa não teria sido possível sem a sua sabedoria, os recursos que disponibilizou e, lógico, as suas cobranças nos momentos certos também.

Ao meu amigo, primeiro orientador e coorientador deste trabalho, Timótiu, eu agradeço por ter me apresentado a robótica e por todos estes anos em que dividimos vitórias, derrotas, alegrias, tristezas, festas e viagens. Como disse nos agradecimentos da monografia, ainda estou te devendo algumas viagens!

Aos meus amigos e companheiros de jornada do LAI (Diego, Harrison, Valentim, Mauro, Anthony e Daniel) e do NERO, ficam os meus mais profundos agradecimentos. Vocês tornaram essa jornada muito mais fácil e prazerosa. Espero ter contribuído com vocês pelo menos uma fração do que contribuíram comigo.

Agradeço também a todos os outros professores e servidores do Programa de Pós-Graduação em Engenharia Elétrica da UFES, assim como aos professores do Instituto de Automática da Universidad Nacional de San Juan pelo suporte à realização da minha pesquisa para Dissertação de Mestrado.

Por fim, agradeço ao CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, agência de fomento do Ministério da Ciência, Tecnologia, Inovação e Comunicações, assim como à FAPES - Fundação de Amparo à Pesquisa e Inovação do Espírito Santo, agência estadual de fomento à pesquisa científica e tecnológica, pelo apoio financeiro que disponibilizou os equipamentos e materiais usados nesta pesquisa. Em particular também agradeço à FAPES pela bolsa de estudos que me foi outorgada, a qual me permitiu dedicar-me exclusivamente ao Mestrado.

*Vinícius Pacheco Bacheti*



*“O futuro pode prometer uma coisa e uma coisa apenas:  
surpresas.”*

***Steven Erikson***



# Resumo

Esta Dissertação de Mestrado discute a realização de uma tarefa de seguimento de caminho por uma formação composta por um veículo terrestre não tripulado (VTNT) e um veículo aéreo não tripulado (VANT). A ideia por trás de tal aplicação é que o VANT está voltando após a entrega de um pacote e deve pousar no VTNT, sua base de referência, que pode estar parado ou seguindo sua rota em uma rua ou estrada. Um controlador de alto nível é projetado para guiar a formação, baseado no paradigma de estruturas virtuais, o qual também inclui um algoritmo de desvio de obstáculos para levar em conta a possibilidade de que o VTNT, a referência para a formação, encontre um obstáculo em seu caminho, como um outro veículo, por exemplo. Após uma descrição detalhada do controlador proposto e do algoritmo de desvio de obstáculos, resultados de simulações e de experimentos são apresentados e discutidos, os quais validam o controlador e o algoritmo de desvio de obstáculos propostos.

**Palavras-chave:** Sistemas multirrobo, Controle de Formação, VANT, Seguimento de caminho, Robôs móveis, Robótica.



# Abstract

This MSc thesis discusses the accomplishment of a path-following task by a formation of an unmanned aerial vehicle (UAV) and an unmanned ground vehicle (UGV). The idea undergoing such application is that the UAV is coming back after delivering a package, and should land on the UGV, its reference base, which can be stopped or continuing to follow its route in a street or a road. A high-level controller is designed to guide the formation, based on the virtual structure paradigm, and includes an obstacle avoidance algorithm to take into account the possibility that the UGV, the reference for the formation, faces an obstacle in its path, such as another vehicle, for instance. After describing the proposed controller and the obstacle avoidance algorithm in details, simulated and experimental results are shown and discussed, which validate the proposed controller and obstacle avoidance algorithm.

**Keywords:** Multi-robot systems, Formation control, UAV, Path-following, Mobile robots, Robotics.



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>23</b>
1.1	Estado da Arte	25
1.2	Motivação	27
1.3	Objetivos	27
1.4	Estrutura do Trabalho	27
<b>2</b>	<b>SEGUIMENTO DE CAMINHO POR MÚLTIPLOS ROBÔS</b>	<b>29</b>
2.1	Robôs Utilizados	29
2.1.1	Modelagem do VTNT	29
2.1.2	Modelagem do VANT	30
2.2	Caracterização da Formação	32
2.3	Definição do Caminho	34
2.4	Controlador de Seguimento de Caminho	35
2.4.1	Sistema Multicamadas de Controle	36
2.4.2	Controle de Orientação do VANT	38
2.4.3	Compensador Dinâmico	39
2.4.4	Estratégia de Redução de Velocidade	39
2.5	Controlador Baseado em Espaço Nulo	40
<b>3</b>	<b>DESVIO DE OBSTÁCULOS</b>	<b>43</b>
3.1	Desvio de um Único Obstáculo	43
3.2	Desvio de Múltiplos Obstáculos	46
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>49</b>
4.1	Seguimento de Caminho e Pouso do VANT sobre o VTNT	49
4.1.1	Pouso Estático	50
4.1.2	Pouso em Movimento	53
4.2	Seguimento de Caminho e Pouso do VANT sobre o VTNT Utilizando Espaço Nulo	57
4.2.1	Pouso Estático	58
4.2.2	Pouso em Movimento	61
4.3	Seguimento de Caminho e Pouso do VANT sobre o VTNT com Obstáculos	62
4.3.1	Simulações	63
4.3.2	Experimentos	66
<b>5</b>	<b>CONCLUSÃO</b>	<b>71</b>
	<b>REFERÊNCIAS</b>	<b>73</b>



# Lista de ilustrações

Figura 1 – Tipos de robôs classificados de acordo com sua locomoção . . . . .	23
Figura 2 – Descrição gráfica da cinemática do robô <i>Pioneer 3-DX</i> . . . . .	30
Figura 3 – Modelo de forças existentes no robô <i>Bebop 2</i> . . . . .	31
Figura 4 – Caracterização da formação composta pelo VTNT e pelo VANT. . . . .	33
Figura 5 – Exemplo das variáveis de caminho. . . . .	35
Figura 6 – Estrutura do controlador responsável por guiar a formação VTNT-VANT no seguimento de um caminho especificado. . . . .	36
Figura 7 – Estrutura de controle adotada, incluindo o uso da técnica de espaço nulo, dando prioridade maior à manutenção da forma da formação. . . . .	42
Figura 8 – Caracterização do desvio de obstáculos, considerando um único obstáculo. . . . .	44
Figura 9 – Situação de desvio de obstáculos considerando o novo sistema de referência, cuja origem é $O_f$ . . . . .	46
Figura 10 – Desvio de obstáculos considerando múltiplos obstáculos próximos uns dos outros. . . . .	47
Figura 11 – Uma visão geral do equipamento de hardware usado para rodar o experimento. . . . .	49
Figura 12 – Foto dos robôs utilizados, assim como do sistema de captura de movimentos <i>OptiTrack</i> . . . . .	50
Figura 13 – Caminho seguido pela formação quando $V_d = 0,45\text{ m/s}$ . . . . .	51
Figura 14 – Erros de formação para a tarefa de seguimento de caminho quando $V_d = 0,45\text{ m/s}$ . . . . .	52
Figura 15 – Velocidades do VTNT para a tarefa de seguimento de caminho quando $V_d = 0,45\text{ m/s}$ . . . . .	52
Figura 16 – Caminho seguido pela formação quando $V_d = 0,5\text{ m/s}$ . . . . .	53
Figura 17 – Caminho percorrido pela formação quando $V_{d_{max}} = 0,45\text{ m/s}$ , com a estratégia de mudar $V_d$ ao longo do caminho. . . . .	54
Figura 18 – Erros de formação quando corrigindo a velocidade no caminho, $V_{d_{max}} = 0,45\text{ m/s}$ . . . . .	54
Figura 19 – Velocidades do VTNT quando corrigindo a velocidade no caminho, $V_{d_{max}} = 0,45\text{ m/s}$ . . . . .	55
Figura 20 – Caminho percorrido pela formação quando $V_{d_{max}} = 0,65\text{ m/s}$ , com a estratégia de mudar $V_d$ ao longo do caminho. . . . .	55
Figura 21 – Caminho percorrido pela formação durante o experimento, dividido em três partes. . . . .	56
Figura 22 – Erros de formação quando o VANT pousa e decola de um VTNT em movimento. . . . .	57
Figura 23 – Velocidades do VTNT para a tarefa de seguimento de caminho quando o VANT pousa e decola de um VTNT em movimento. . . . .	58
Figura 24 – Caminho percorrido pela formação quando há prioridade de forma. . . . .	59
Figura 25 – Erros de formação quando há prioridade de forma. . . . .	60
Figura 26 – Velocidades do VTNT quando há prioridade de forma. . . . .	60
Figura 27 – Caminho percorrido pela formação durante o experimento, dividido em três partes. . . . .	61
Figura 28 – Erros de formação para prioridade de forma. . . . .	62
Figura 29 – Velocidades do VTNT com prioridade de forma. . . . .	62
Figura 30 – Diferentes cenários imaginados para testar o algoritmo de desvio de obstáculos associado ao controlador de seguimento de caminho proposto. . . . .	64
Figura 31 – Caminho realizado pela formação durante a navegação, dividido em três partes. . . . .	67
Figura 32 – Erros de formação. . . . .	68
Figura 33 – Velocidades linear e angular. . . . .	69



# Lista de tabelas

Tabela 1 – Valores dos parâmetros do modelo identificados. . . . .	32
--	----



# Lista de Abreviaturas

VANT	Veículo Aéreo Não Tripulado
VTNT	Veículo Terrestre Não Tripulado
ROS	do Inglês, <i>Robot Operating System</i>



# 1 Introdução

Robôs autônomos são aqueles que têm capacidade de determinar quais ações devem ser tomadas, usando algum tipo de sistema de percepção e de cognição (RUBIO; VALERO; LLOPIS-ALBERT, 2019). Eles têm vasta gama de aplicações, que vão desde uso industrial até exploração planetária, passando por sistemas de vigilância, uso em construções, uso como guias de museus, uso em cuidados médicos, intervenções em ambientes extremos e muitas outras. Robôs móveis podem ser distinguidos de outros tipos de robôs pela sua capacidade de se mover de maneira autônoma, com inteligência o suficiente para reagir e tomar decisões com base em sua percepção do ambiente ao seu redor.

A robótica móvel consiste basicamente dos campos de locomoção, percepção, cognição e navegação (SIEGWART, 2004). Problemas de locomoção são resolvidos através de análises cinemáticas, dinâmicas e de teoria de controle. Percepção, por sua vez, envolve as áreas de análise de sinais e áreas específicas como visão computacional e sensores. A cognição é responsável por analisar dados fornecidos pelos sensores e tomar as ações necessárias para atingir os objetivos da tarefa. Aqui se enquadram os sistemas de controle do robô. A navegação é responsável por algoritmos de planejamento da rota.

Uma das classificações de robôs móveis é feita de acordo com o meio no qual eles se locomovem. Robôs terrestres muitas vezes são chamados de veículos terrestres não tripulados (VTNT) enquanto robôs que se locomovem pelo ar são chamados de veículos aéreos não tripulados (VANT). A Figura 1 mostra o robô terrestre do tipo unicycle *Pioneer 3-DX*, fabricado pela *Adept Mobile Robots*, como exemplo de um VTNT, e o robô aéreo do tipo quadrimotor *Bebop 2*, fabricado pela *Parrot Drones SAS*, como exemplo de um VANT.



(a) Exemplo de um VTNT



(b) Exemplo de um VANT

Figura 1 – Tipos de robôs classificados de acordo com sua locomoção

Normalmente as tarefas de navegação são divididas em posicionamento, seguimento de trajetória e seguimento de caminho. Nas tarefas de posicionamento, apenas a posição final desejada do robô é pedida e não se considera o trajeto desenvolvido para atingi-la, e nem a sua velocidade. No seguimento de trajetória existe uma posição (e em alguns casos orientação) variante no tempo, a qual o robô deve atingir. Além disso, a velocidade desejada para cada momento é controlada de acordo com a variação no tempo da trajetória descrita e se por algum motivo o robô não desenvolver tal velocidade, a trajetória continua sua evolução, aumentando assim os erros de posicionamento. O seguimento de caminho implica que o veículo deve seguir um caminho

especificado, porém a velocidade desejada pode ser escolhida livremente, o que aumenta a flexibilidade do sistema.

Veículos aéreos não tripulados têm sido utilizados na execução de diversas tarefas, principalmente em virtude da sua capacidade de coletar imagens com vista do alto (a chamada *bird's eye view* - *vista dos pássaros*, como aplicações em inspeção de infraestruturas (JORDAN et al., 2018; JIANG et al., 2019; SHAKHATREH et al., 2019), inspeções na agricultura (ZHANG; KOVACS, 2012; MURUGAN; GARG; SINGH, 2017) ou mapeamento de áreas (KIM; KWON; SEO, 2014; LYU et al., 2015; QIN et al., 2019), enquanto outras não necessariamente usam imagens, como aplicações de transporte de carga (PALUNKO; CRUZ; FIERRO, 2012; DAI; LEE; BERNSTEIN, 2014; PIZETTA; BRANDÃO; SARCINELLI-FILHO, 2020; MEISSEN et al., 2017; SHIRANI; NAJAFI; IZADI, 2019; VILLA; BRANDÃO; SARCINELLI-FILHO, 2019). De fato, mudanças nas regras em alguns países têm aberto a possibilidade de operações comerciais com uso de drones, como são popularmente conhecidos os VANTs, aumentando sua utilização no setor civil (SOLOVEY; RYAN; CUMMINGS, 2020). Desse modo, a interação entre veículos aéreos e terrestres tem sido alvo de pesquisa por cientistas do mundo inteiro, já que tal estratégia permite a superação de limites intrínsecos a cada robô, ampliando o leque de aplicações.

Equipes envolvendo múltiplos VANTs ou grupos de VANTS e VTNTs têm sido utilizadas em diversas tarefas, de maneira a obter mais eficiência, em termos de tempo de cumprimento da tarefa ou de custo-benefício, já que um time de robôs genéricos pode ser mais barato que um único robô especializado (PARKER, 2016; RABELO; BRANDÃO; SARCINELLI-FILHO, 2020). Outras vezes, a natureza da tarefa pede um time de robôs. Como exemplo de tarefas com múltiplos robôs pode-se citar as aplicações em (DONG et al., 2019; PIZETTA; BRANDÃO; SARCINELLI-FILHO, 2019; MASONE; BÜLTHOFF; STEGAGNO, 2016; BAI; WEN, 2010; TOKEKAR et al., 2016; VILLA; BRANDÃO; SARCINELLI-FILHO, 2019), entre outras. Além disso, como visto em (BACHETI; BRANDÃO; SARCINELLI-FILHO, 2020) o uso de múltiplos robôs viabiliza a redundância de informação, podendo garantir a execução da tarefa mesmo diante da falha de algum agente, como é o caso em (ERNANDES-NETO; SARCINELLI-FILHO; BRANDÃO, 2019), ou da diversidade no ambiente de atuação, como em (SHARMA; KUMAR, 2015).

O paradigma de controle de formação chamado de estrutura virtual (LEWIS; TAN, 1997) consiste em uma figura geométrica que descreve a forma da formação e sua dimensão. Exemplos de tais estruturas são um triângulo, como em (KITTS; MAS, 2009a), e uma reta, como em (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020). Para a navegação da formação adota-se um ponto de referência, que pode ser o centroide da estrutura virtual (KITTS; MAS, 2009a) ou mesmo uma extremidade de tal estrutura (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020). Neste último caso um ponto conveniente é a extremidade da linha reta que coincide com o robô terrestre, quando se utiliza uma formação VTNT-VANT, como é o caso explorado nesta Dissertação de Mestrado. Um robô pontual e sem massa é colocado em tal ponto, o qual, ao mover-se, move toda a formação. Além disso, de maneira a permitir o pouso do VANT sobre o VTNT, controla-se a forma da estrutura linear de maneira que se garanta que o VANT chegue perto o suficiente do VTNT para pousar. Portanto, a tarefa de mover a formação seguindo um caminho pré-estabelecido consiste em fazer com que o robô virtual que a caracteriza siga o caminho desejado e, ao mesmo tempo, mantendo a forma da estrutura virtual, a qual é um reflexo das posições dos robôs que a compõem. Portanto, podem-se considerar, basicamente, duas tarefas parciais (MOREIRA; BRANDÃO; SARCINELLI-FILHO, 2019): controlar a posição e a forma da estrutura virtual (e assim da formação).

Quando diferentes tarefas entram em contradição, faz-se necessário o uso de técnicas que permitam hierarquizá-las, por meio das prioridades a elas atribuídas. O uso do Espaço Nulo, como visto em (ANTONELLI; ARRICHELLO; CHIAVERINI, 2008), permite tal hierarquização de maneira analítica, evitando a comutação

entre diferentes controladores. Em tal contexto, este trabalho utiliza a técnica de espaço nulo e define diferentes prioridades para as tarefas de mover a formação e manter a sua forma, considerando uma formação composta por um VANT e um VTNT.

Em relação à percepção dos robôs, o nível de sofisticação requerido depende do ambiente na qual eles se encontram. Ambientes estruturados são aqueles em que a localização dos objetos contidos nele são completamente conhecidas e não há modificações físicas no espaço, de sorte que a percepção do robô sobre o mundo externo pode ser mais simples, já que este é previsível. Em um ambiente semiestruturado, que é o caso aqui considerado, há conhecimento parcial do ambiente, mas ele pode se modificar com o tempo. Um exemplo é a navegação em uma estrada, em que se conhece o trajeto e se sabe que se trata de um piso plano, mas condições como outros carros na pista e sinalização interferem na navegação. Já o ambiente não estruturado é um ambiente completamente desconhecido (TZAFESTAS, 2014).

Em um ambiente semiestruturado, onde há expectativas de obstáculos, um algoritmo que garanta o desvio dos mesmos é essencial para a navegação segura do robô ou da formação. Neste contexto, diversas abordagens podem ser adotadas. A mais simples delas é construir um mapa do ambiente, marcar os obstáculos e usar um dos algoritmos de planejamento de caminhos, tais como os algoritmos de Dijkstra, A\* (DAS; KONAR; LAISHRAM, 2010) e de Bellman-Ford, entre outros (KWAŚNIEWSKI; GOSIEWSKI, 2018). Infelizmente, tais algoritmos requerem alto esforço computacional, e muitas vezes o conhecimento que o robô tem do ambiente pode não ser suficiente para usar tais abordagens. Outras abordagens comuns são o método de campos potenciais (SANTOS et al., 2017; KOVÁCS et al., 2016; CERQUEIRA; SANTOS; CONCEICAO, 2016), que conta com uma análise matemática elegante, métodos fuzzy-neurais (LIU; LIU; WANG, 2016; POTHAL; PARHI, 2015) e algoritmos genéticos (KWAŚNIEWSKI; GOSIEWSKI, 2018; MATHIAS; RAGUSA, 2016). Entretanto, uma desvantagem encontrada nos métodos de campos potenciais e em algoritmos genéticos é a possibilidade de que os robôs fiquem presos em mínimos locais o que ocorre quando eles navegam em ambientes com obstáculos muito próximos ou passagens estreitas (YAN et al., 2018).

## 1.1 Estado da Arte

Conforme mencionado, o uso de VANTs, pela sua capacidade de coleta de imagens, é um tópico relevante atualmente. O estado corrente do uso de VANTs em inspeções é abordado por (JORDAN et al., 2018), que justifica seu uso através de reduções de custo em equipamentos especializados e interrupções de trabalho enquanto a inspeção manual, a forma antiga, é realizada. As principais dificuldades enfrentadas, como estabilidade de voo, exatidão do controle e segurança, são discutidas, passando pelas tecnologias que estão permitindo a solução destes problemas e os desafios atuais e futuros. Um desses desafios na inspeção por imagens é discutido em (JIANG et al., 2019). As câmeras dos VANTs normalmente possuem um sistema de estabilização, por software ou hardware, para que mudanças na atitude do veículo não distorçam as imagens. Atrasos nesse sistema, porém, podem fazer com que ruídos vindos dessas mudanças de angulação ainda sejam problemáticos. Uma solução baseada no controle dos vetores de impulso é proposta de maneira que o veículo possa se movimentar sem alterar sua atitude, levando a imagens mais fiéis.

Já na indústria agrícola dados de satélites têm sido usados há décadas para classificação em larga escala, e em anos recentes drones para inspeções precisas em pequena escala vêm sendo utilizados. Porém, como visto em (MURUGAN; GARG; SINGH, 2017), classificações precisas em grande escala são ainda um desafio. A solução proposta junta dados de satélites e drones e, como o uso repetido do VANT deve ser evitado, uma abordagem adaptativa de classificação é usada, através de estatísticas de imagens da região selecionada. No setor civil, conforme comentado em (SOLOVEY; RYAN; CUMMINGS, 2020), há um grande esforço no sentido

de desenvolver interfaces que simplifiquem e deixem mais seguro o uso de VANTs por pessoas com pouco treino, de maneira a permitir o avanço dessa tecnologia em escalas menores neste setor.

No contexto de formações de vários robôs, carregamento de carga é um tema comum para grupos de VANT. Por exemplo, o trabalho em (MEISSEN *et al.*, 2017) apresenta um controlador de formação baseado em passividade para carregamento de uma carga pontual. Os autores atingem tal objetivo usando um controlador interno para cada veículo e uma lei de controle de formação que rege a posição relativa entre os robôs. Em (PIZETTA; BRANDÃO; SARCINELLI-FILHO, 2019) o problema tratado, uma formação de três drones carregando uma carga pontual, é parecido, porém ali a solução proposta é uma lei de controle não linear baseada em linearização por realimentação, e os autores mostram a formação executando tarefas de posicionamento e seguimento de trajetória. O controlador é responsável por manter a formação estável enquanto compensa os distúrbios causados pela carga e pelas tensões nos cabos. Já em (VILLA; BRANDÃO; SARCINELLI-FILHO, 2019) a carga analisada é em formato de barra, e se utilizam apenas dois drones. É válido comentar que as duas últimas estratégias discutidas utilizam o conceito de formação por estruturas virtuais, a mesma abordagem que será utilizada nesta dissertação de mestrado.

Uma estratégia de cooperação entre VANTs e VTNTs para mapeamento de regiões onde não há GPS disponível é proposta em (QIN *et al.*, 2019). O sistema utilizado implementa uma estratégia de exploração em duas etapas. A tarefa de percepção é realizada primeiramente em uma camada de exploração mais grosseira, feita pelo VTNT usando a técnica de localização e mapeamento simultâneos (SLAM). Essa etapa serve então de referência para o VANT executar uma camada mais detalhada de mapeamento. O sistema utiliza como vantagem a agilidade do VANT e o grande poder computacional que pode ser colocado a bordo do VTNT.

Um conceito de formação de robôs chamado de contenção da formação é discutido em (DONG *et al.*, 2019). Tal conceito admite a existência de múltiplos líderes, e as variáveis de estado dos seguidores devem permanecer no envoltório convexo criado pelos líderes.

Os trabalhos expostos em (MOREIRA; BRANDÃO; SARCINELLI-FILHO, 2019) e (ERNANDES-NETO; SARCINELLI-FILHO; BRANDÃO, 2019) usam a técnica de espaço nulo para fazer o controle hierárquico de sub tarefas de uma formação heterogênea. As tarefas são a navegação do elemento virtual da formação e a manutenção da sua forma. Esta técnica funciona projetando as ações da tarefa de menor prioridade no espaço nulo da matriz Jacobiana que caracteriza a tarefa de maior prioridade. Desta forma, para a tarefa de menor prioridade, as ações de controle só serão tomadas se não entrarem em conflito com a tarefa de maior prioridade.

Em (PIZETTA; BRANDÃO; SARCINELLI-FILHO, 2020) uma estratégia de desvio de obstáculos considerando campos potenciais é desenvolvida no contexto de uma aplicação de resgate na floresta, onde o veículo aéreo carrega uma carga de suprimentos médicos. Nesse sentido, o VANT deve se desviar de troncos de árvore espalhados no caminho, enquanto a carga é considerada uma perturbação modelada. Já em (SANTOS *et al.*, 2017), a técnica de campos potenciais também é utilizada, porém os autores usam o espaço nulo como forma de adicionar a tarefa de desvio de obstáculos como prioritária em relação à tarefa de navegação. Finalmente, usando como pano de fundo explorações interplanetárias, que oferecem um ambiente difícil e muitas vezes desconhecido, em (KWAŚNIEWSKI; GOSIEWSKI, 2018) se discute uma solução de desvio de obstáculos por algoritmos genéticos. Enquanto métodos gráficos utilizam alto poder computacional e métodos como redes neurais e controladores *fuzzy* têm baixa eficiência em ambientes complexos, os autores mostram que o algoritmo proposto completou com sucesso as manobras de desvio nos mapas propostos, incluindo situações clássicas de desvio como becos sem saída.

## 1.2 Motivação

Nas cadeias de fornecimento de produtos, a logística da última milha, que corresponde à chegada ao destinatário e entrega do pacote, é o estágio menos eficiente, e sozinho pode ser responsável por até 28% do total do custo de entrega (RANIERI et al., 2018). Mais ainda, o alto uso de veículos na cidade aumenta o tráfego diminuindo dramaticamente a eficiência do sistema de transporte, o que causa perda de tempo e dinheiro. Por fim, o trânsito e estacionamento de veículos, especialmente de médio e grande porte, são responsáveis pela ocupação de espaço e danos à infraestrutura da cidade.

Com isto em mente, este trabalho aborda o problema da navegação conjunta de uma formação composta por um VTNT e um VANT (formação heterogênea) no seguimento de caminhos, com a possibilidade do VANT pousar ou decolar sobre o VTNT. Isto pode ser aplicado à uma tarefa de entrega de pacotes por drones, onde o veículo aéreo decola de uma base móvel, como por exemplo um caminhão de entregas, quando estiver perto do endereço final. Ao terminar a entrega, o drone volta ao caminhão e pousa sobre ele. Desta maneira, o caminhão poderia navegar apenas pelas vias principais da cidade, enquanto os veículos aéreos são responsáveis pelas entregas aos endereços finais. Neste trabalho, a abordagem utilizada consiste em considerar os dois veículos como uma formação de robôs e, a partir disso, controlar a navegação de ambos os veículos de maneira coordenada.

Portanto, ao se considerar uma tarefa de entrega em que um drone pousa em um caminhão, onde restrições naturais a estradas e rodovias, como semáforos e outros carros, podem afetar a velocidade desejada de maneira imprevisível, uma aplicação de seguimento de caminho é a mais recomendada, se comparada com um seguimento de trajetória.

Para lidar com as restrições citadas no parágrafo anterior e superar algumas das limitações citadas nos algoritmos de desvio de obstáculos, o presente trabalho propõe uma solução reativa de planejamento de caminhos para desvio de obstáculos, considerando que o robô virtual que caracteriza a formação não tem conhecimento a priori dos obstáculos, mas os detecta durante a navegação.

## 1.3 Objetivos

O objetivo geral desta Dissertação de Mestrado é desenvolver um controlador de formação que permita o cumprimento da tarefa de seguimento de caminho e pouso de um VANT sobre um VTNT, simulando a última etapa de uma tarefa de entrega de pacotes. Além disso, como objetivos secundários, serão analisados os impactos da aplicação da técnica de espaço nulo para hierarquização das sub-tarefas de navegação e será desenvolvido um algoritmo de desvio de obstáculos, assim como sua validação por meio de simulações e experimentos.

## 1.4 Estrutura do Trabalho

A organização da dissertação se dá na seguinte estrutura:

### **Capítulo 1: Introdução**

Neste capítulo é feito um breve resumo dos conceitos tratados nesta dissertação, junto com o estado da arte, assim como se apresentam a motivação do trabalho e o objetivo que se pretende atingir.

### **Capítulo 2: Seguimento de Caminho por Múltiplos Robôs**

Neste capítulo são descritos de maneira mais detalhada o tipo de formação usada neste trabalho, as

definições de caminho e de que maneira se constroem os controladores. Além disso, se desenvolve a aplicação de espaço nulo.

### Capítulo 3: Desvio de Obstáculos

Neste capítulo a construção do algoritmo de desvio de obstáculos para um único robô é detalhada, e posteriormente é estendida para lidar com múltiplos robôs.

### Capítulo 4: Resultados e Discussões

Neste capítulo os resultados obtidos em simulações e experimentos são expostos e discutidos de maneira detalhada, mostrando-se que os mesmos validam o sistema proposto.

### Capítulo 5: Conclusão

Este capítulo elenca as principais conclusões do trabalho realizado, assim como propostas para continuação da pesquisa.

Por fim, vale destacar que a pesquisa realizada para gerar esta Dissertação de Mestrado gerou algumas publicações, a saber:

1. Vinicius Pacheco Bacheti, Alexandre Santos Brandão e Mário Sarcinelli Filho, “Path-Following with a UGV-UAV Formation Considering that the UAV Lands on the UGV”, 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 2020, pp. 488-497, doi: 10.1109/ICUAS48674.2020.9213918
2. Vinicius Pacheco Bacheti, Alexandre Santos Brandão e Mário Sarcinelli Filho, “Seguimento de Caminho Usando Espaço Nulo para uma Formação VANT-VTNT”, XXIII Congresso Brasileiro de Automática (CBA2020), Porto Alegre, RS, 2020, doi: 10.48011/asba.v2i1.1478
3. Anthony Oliveira Pinto, Harrison Neves Marciano, Vinicius Pacheco Bacheti, Mauro Sergio Mafra Moreira, Alexandre Santos Brandão e Mário Sarcinelli-Filho, “High-Level Modeling and Control of the Bebop 2 Micro Aerial Vehicle”, 2020 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 2020, pp. 939-947, doi: 10.1109/ICUAS48674.2020.9213941
4. Harrison Neves Marciano, Anthony Oliveira Pinto, Vinicius Pacheco Bacheti, Mauro Sergio Mafra Moreira, Alexandre Santos Brandão e Mário Sarcinelli-Filho, “Modelagem de Alto Nível e Controle do Quadrimotor Bebop 2”, XXIII Congresso Brasileiro de Automática (CBA2020), Porto Alegre, RS, 2020, doi: 10.48011/asba.v2i1.1114

Além de tais publicações já efetivadas, um outro artigo está sendo preparado para ser submetido à revista Control Engineering Practice.

## 2 Seguimento de Caminho por Múltiplos Robôs

Neste capítulo são abordados os modelos dos robôs utilizados, assim como de que maneira se caracterizam a formação e o caminho. O controlador é então formalizado e, por fim, os conceitos de espaço nulo e como eles se encaixam na estrutura de controle são discutidos.

### 2.1 Robôs Utilizados

Nesta seção são descritos os robôs utilizados para compor a formação, assim como seus modelos. O VTNT escolhido foi o *Pioneer 3-DX*, enquanto o VANT usado foi o *Bebop 2*.

#### 2.1.1 Modelagem do VTNT

O *Pioneer 3-DX* é um robô terrestre do tipo uniciclo que conta com duas rodas com tração independentes e uma roda castor para suporte. Devido à sua arquitetura ele não consegue se movimentar na direção do eixo que liga suas duas rodas, ou seja, apresenta uma restrição não holonômica. Todos os movimentos do robô podem ser então decompostos em dois tipos de velocidade: a velocidade linear  $v$ , que tem a mesma direção da frente do robô, e a velocidade angular  $\omega$  cujo sentido positivo é o anti-horário. Desta forma, define-se

$$\mathbf{u} = \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2.1)$$

como o vetor de velocidades instantâneas do veículo em seu próprio sistema de referência (eixos  $X^r$  e  $Y^r$ , mostrados na Figura 2). Já as velocidades do robô no referencial do mundo (eixos  $X^w$  e  $Y^w$ , também mostrados na Figura 2) são dadas por

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix}. \quad (2.2)$$

A relação entre as velocidades  $\mathbf{u}$  e  $\dot{\mathbf{x}}$  é dada por

$$\dot{\mathbf{x}} = \mathbf{K}_p \mathbf{u} \quad (2.3)$$

onde  $\mathbf{K}_p$  é chamada de matriz de cinemática direta. O formato de  $\mathbf{K}_p$  depende do ponto de controle escolhido. Neste trabalho, como mostra a Figura 2, foi escolhido um ponto de controle localizado à frente do centro do robô, a uma distância  $a$  do eixo virtual que une as duas rodas de tração. Neste caso,

$$\mathbf{K}_p = \begin{bmatrix} \cos \psi & -a \sin \psi \\ \sin \psi & a \cos \psi \end{bmatrix} \quad (2.4)$$

onde  $\psi$  é o ângulo entre os eixos do robô e os eixos correspondentes do mundo.

A relação contrária, ou seja, que converte as velocidades do referencial global para o do robô, é chamada de cinemática inversa, e assume a forma

$$\mathbf{u} = \mathbf{K}_p^{-1} \dot{\mathbf{x}} \quad (2.5)$$

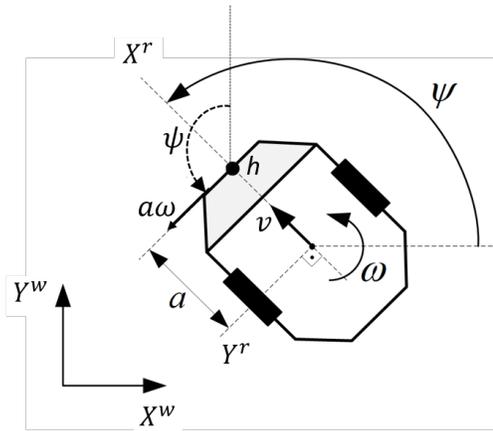


Figura 2 – Descrição gráfica da cinemática do robô *Pioneer 3-DX*

onde

$$\mathbf{K}_p^{-1} = \begin{bmatrix} \cos \psi & \text{sen} \psi \\ -\frac{\text{sen} \psi}{a} & \frac{\cos \psi}{a} \end{bmatrix} \quad (2.6)$$

é a matriz de cinemática inversa, que é a inversa da matriz de cinemática direta.

### 2.1.2 Modelagem do VANT

O *Bebop 2* é um veículo aéreo do tipo quadrimotor originalmente projetado para ser controlado por *smartphones* ou *tablets*, geralmente para propósito de entretenimento ou uso de sua câmera. Entretanto, devido à implementação de uma biblioteca disponível em ROS<sup>1</sup>, cuja sigla significa *Robot Operating System*, ou Sistema Operacional de Robôs, em português, é possível realizar a conexão de um computador externo com o sistema interno do robô, e dessa maneira usá-lo como uma plataforma de experimentos controlada a partir de um computador.

Em relação ao modelo adotado aqui, o *Bebop 2* tem quatro asas rotativas, e seu movimento é baseado na resposta a mudanças na velocidade de cada rotor. Como consequência de tais mudanças, quatro forças surgem, uma associada a cada rotor, e ao inclinar o veículo girando-o ao redor de algum dos eixos do seu sistema de referência (eixos  $x^b$ ,  $y^b$  e  $z^b$  mostrados na Figura 3) são geradas forças, e assim acelerações, nas três direções do espaço 3D cartesiano (eixos  $x^w$ ,  $y^w$  e  $z^w$ , também mostrados na Figura 3), e torques em torno de cada um desses eixos. A combinação de tais forças e torques faz com que o veículo voe em qualquer direção desejada, com a velocidade desejada, desde que tal velocidade esteja dentro dos limites impostos pelo fabricante. A Figura 3, encontrada em (VILLA; BRANDÃO; SARCINELLI-FILHO, 2020), mostra tais forças e graus de liberdade do VANT (velocidades lineares nas três direções do sistema cartesiano e rotação em torno de cada um dos seus eixos).

Um aspecto importante do *Bebop 2* é que ele vem equipado com um piloto automático, que nada mais é que um controlador de baixo nível responsável por controlar a atitude do veículo. Tal piloto automático permite que o usuário limite os ângulos de rolagem (rotação ao redor do eixo  $x^b$ ) e arfagem (rotação ao redor do eixo  $y^b$ ),  $\phi$  e  $\theta$ , respectivamente. Além disso, também podem ser limitadas as velocidades lineares e angulares referentes ao eixo  $z$  do robô. Em aplicações nas quais não há necessidade de manobras agressivas, o

<sup>1</sup> Veja <https://www.ros.org/>

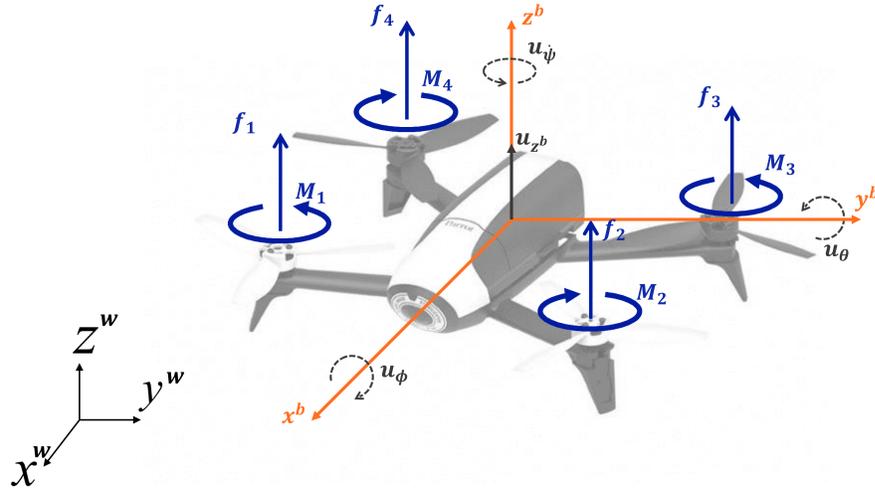


Figura 3 – Modelo de forças existentes no robô *Bebop 2*.

usuário pode limitar os ângulos de rolagem e arfagem em valores pequenos, permitindo a aproximação de tais valores a zero, o que significa que a plataforma correspondente ao corpo do quadrimotor está sempre na posição horizontal, mas ainda assim tem capacidade de desenvolver velocidades nas direções  $x$  e  $y$  do espaço 3D (SANTANA et al., 2014; SANTANA; BRANDÃO; SARCINELLI-FILHO, 2016; PINTO et al., 2020). Isso permite o envio de comandos no formato de um vetor com quatro posições, que são as três velocidades lineares nos três eixos  $x^b$ ,  $y^b$  e  $z^b$  e a velocidade angular em torno do eixo  $z^b$ , chamada de velocidade de guinada.

O vetor de comandos é então definido como

$$\mathbf{u}_d = \begin{bmatrix} u_{d_{v_x}} \\ u_{d_{v_y}} \\ u_{d_z} \\ u_{d_{\dot{\psi}}} \end{bmatrix}, \quad (2.7)$$

onde  $u_{d_{v_x}}$  e  $u_{d_{v_y}}$  são os comandos de velocidade nos eixos  $x^b$  e  $y^b$ ,  $u_{d_z}$  é o comando de velocidade no eixo  $z^b$  e  $u_{d_{\dot{\psi}}}$  é comando de velocidade de guinada.

Todos estes comandos são enviados ao veículo aéreo como valores normalizados, indo de 1, que corresponde aos valores máximos de ângulo de arfagem e rolagem, a velocidade vertical máxima e velocidade de guinada máxima nas direções positivas, até  $-1$ , que corresponde a esses mesmos valores, porém na direção contrária. Conforme mencionado, tais parâmetros são configuráveis através do auto piloto interno do *Bebop 2*. No presente trabalho, os valores máximos de ângulo de arfagem e rolagem são de  $5^\circ$ , a velocidade vertical máxima é de  $1m/s$  e a velocidade de guinada máxima é de  $100^\circ/s$ . Em relação às suas direções positivas, estas são, respectivamente, frontal, para a esquerda, para cima e anti-horária (ver Figura 3).

Como consequência de tais simplificações, o modelo dinâmico simplificado proposto em (SANTANA et al., 2014; PINTO et al., 2020) é adotado para representar o veículo, e é escrito como

$$\ddot{\mathbf{x}} = \mathbf{f}_1 \mathbf{u}_d - \mathbf{f}_2 \dot{\mathbf{x}}. \quad (2.8)$$

Este modelo foi demonstrado como uma aproximação aceitável para a dinâmica do veículo quando este apresenta valores baixos de ângulos de arfagem e rolagem (condição de voo quase pairado). Ele se baseia na suposição que o veículo tem um auto piloto embutido que cuida da estabilização do veículo em baixos valores angulares o que, como explicado previamente, é verdadeiro no veículo usado neste trabalho. Tal modelo é uma

representação de espaço de estados da dinâmica do quadrimotor, onde os estados, que estão no vetor  $\dot{\mathbf{x}}$ , são as velocidades lineares nos três eixos e a velocidade angular em torno do eixo  $z$ , todas na referência do mundo. Em tal modelo o vetor  $\ddot{\mathbf{x}}$  representa a derivada temporal do vetor de estados  $\dot{\mathbf{x}}$ , ou seja

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\psi} \end{bmatrix}, \text{ e } \ddot{\mathbf{x}} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\psi} \end{bmatrix}. \quad (2.9)$$

As matrizes  $\mathbf{f}_1$  e  $\mathbf{f}_2$ , por sua vez, são dadas por

$$\mathbf{f}_1 = \begin{bmatrix} K_1 \cos \psi & -K_3 \sin \psi & 0 & 0 \\ K_1 \sin \psi & K_3 \cos \psi & 0 & 0 \\ 0 & 0 & K_5 & 0 \\ 0 & 0 & 0 & K_7 \end{bmatrix} \text{ e} \quad (2.10)$$

$$\mathbf{f}_2 = \begin{bmatrix} K_2 \cos \psi & -K_4 \sin \psi & 0 & 0 \\ K_2 \sin \psi & K_4 \cos \psi & 0 & 0 \\ 0 & 0 & K_6 & 0 \\ 0 & 0 & 0 & K_8 \end{bmatrix},$$

onde  $\psi$  é o ângulo entre os eixos  $x^b$  e  $x^w$ , o qual define a matriz de cinemática do veículo, a saber,

$$\mathbf{K}_b = \begin{bmatrix} \cos \psi & -\sin \psi & 0 & 0 \\ \sin \psi & \cos \psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.11)$$

a qual mapeia as velocidades nos eixos do robô para os eixos do sistema global de referência.

Já as constantes  $K_1, \dots, K_8$  que aparecem em  $\mathbf{f}_1$  e  $\mathbf{f}_2$  são valores experimentais obtidos usando o sistema de captura de movimento OptiTrack<sup>2</sup>. Os valores utilizados neste trabalho, que podem ser vistos na Tabela 1, são aqueles obtidos em (PINTO et al., 2020), onde está exposto o procedimento para obtenção dos mesmos. É válido ressaltar que os valores máximos dos comandos influenciam nestes valores, por isso foi importante sua definição nos parágrafos anteriores.

Tabela 1 – Valores dos parâmetros do modelo identificados.

$K_1$	$K_2$	$K_3$	$K_4$	$K_5$	$K_6$	$K_7$	$K_8$
0.8417	0.18227	0.8354	0.17095	3.966	4.001	9.8524	4.7295

## 2.2 Caracterização da Formação

A caracterização da formação considerada neste trabalho é baseada no paradigma de estruturas virtuais (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020; LEWIS; TAN, 1997; LOW, 2014), onde a estrutura considerada é a linha 3D que liga os dois veículos. De acordo com tal paradigma, um robô virtual é colocado

<sup>2</sup> Ver <https://optitrack.com/applications/robotics/#intro>

em um ponto de referência no elemento geométrico que caracteriza a formação, o qual pode ser em seu centro de gravidade, como em (KITTS; MAS, 2009b; BRANDÃO et al., 2015; RESENDE; CARELLI; SARCINELLI-FILHO, 2014). No caso da formação VTNT-VANT isto seria o ponto central da linha correspondente à interligação dos veículos. Entretanto, levando em consideração que o objetivo é propor um sistema de controle que permita o pouso do VANT sobre o VTNT, o ponto escolhido como referência para o robô virtual foi a extremidade da linha que coincide com a posição do VTNT, como em (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020). Portanto, a formação é caracterizada como mostrado na Figura 4, com o robô virtual associado à formação sendo o próprio robô *Pioneer 3-DX*, porém sem consideração de sua inércia (por ser um robô virtual).

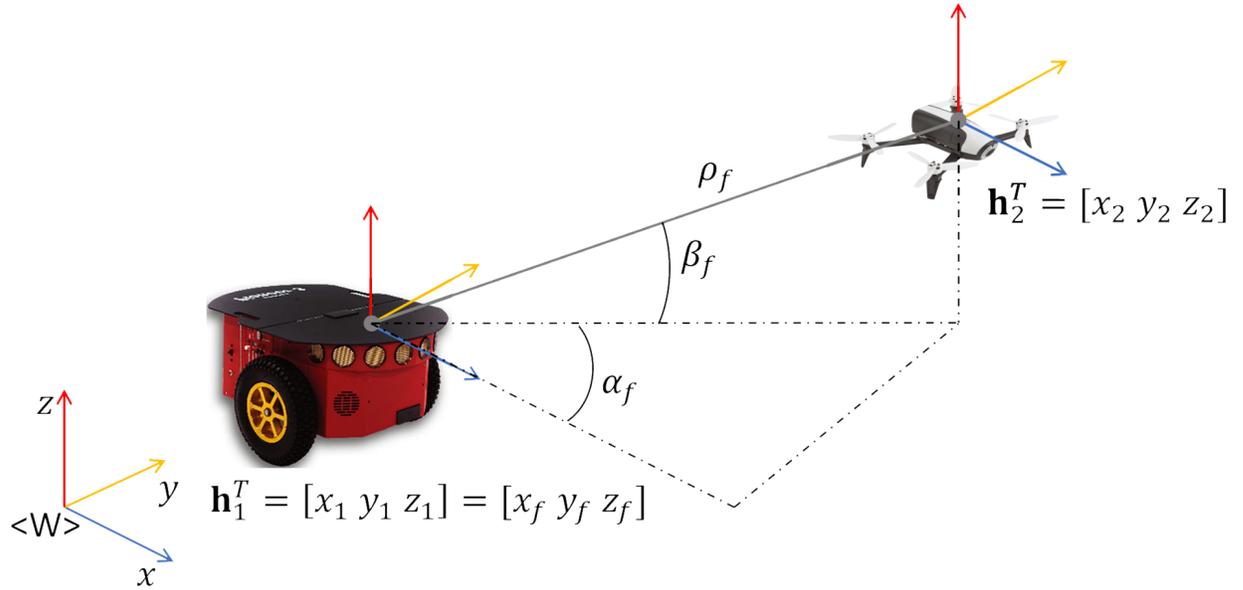


Figura 4 – Caracterização da formação composta pelo VTNT e pelo VANT.

As variáveis que descrevem a formação também são mostradas na Figura 4, com os índices 1 e 2 se referindo ao VTNT e ao VANT, respectivamente. As variáveis correspondentes às posições do VTNT,

$$\mathbf{h}_1 = [x_1 \quad y_1 \quad z_1]^T, \quad (2.12)$$

e do VANT,

$$\mathbf{h}_2 = [x_2 \quad y_2 \quad z_2]^T, \quad (2.13)$$

são agrupadas no vetor único de posição

$$\mathbf{x} = [\mathbf{h}_1^T \quad \mathbf{h}_2^T]^T = [x_1 \quad y_1 \quad z_1 \quad x_2 \quad y_2 \quad z_2]^T. \quad (2.14)$$

Tal vetor descreve as chamadas variáveis dos robôs, enquanto as variáveis agrupadas no vetor

$$\mathbf{q} = [x_f \quad y_f \quad z_f \quad \rho_f \quad \alpha_f \quad \beta_f]^T \quad (2.15)$$

são chamadas de variáveis da formação. Portanto, considerando que o robô virtual coincide com o VTNT, tem-se que  $[x_f \quad y_f \quad z_f]^T = [x_1 \quad y_1 \quad z_1]^T$ . Sendo assim, dois espaços diferentes são definidos, o espaço da

formação ou *cluster* (KITTS; MAS, 2009b; RESENDE et al., 2013), e o espaço dos robôs. Tais espaços são conectados por duas transformações, uma direta e outra inversa. A primeira permite a obtenção de  $\mathbf{q}$  a partir de  $\mathbf{x}$ , ou

$$\mathbf{q} = f(\mathbf{x}), \quad (2.16)$$

onde

$$f(\mathbf{x}) = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \\ \operatorname{tg}^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \\ \operatorname{tg}^{-1}\left(\frac{z_2 - z_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}\right) \end{bmatrix}. \quad (2.17)$$

Já a transformação inversa permite obter  $\mathbf{x}$  a partir de  $\mathbf{q}$ , ou

$$\mathbf{x} = f^{-1}(\mathbf{q}), \quad (2.18)$$

onde

$$f^{-1}(\mathbf{q}) = \begin{bmatrix} x_f \\ y_f \\ z_f \\ x_f + \rho_f \cos(\alpha_f) \cos(\beta_f) \\ y_f + \rho_f \operatorname{sen}(\alpha_f) \cos(\beta_f) \\ z_f + \rho_f \operatorname{sen}(\beta_f) \end{bmatrix}. \quad (2.19)$$

Uma observação importante é que é considerada a existência de uma posição  $z_1$  para o VTNT, que é sempre igual a zero. Isto é feito porque dessa forma o mesmo formalismo pode ser usado para uma formação composta por dois VANTs, como feito em (PINTO et al., 2020).

### 2.3 Definição do Caminho

O objetivo de uma tarefa de seguimento de caminho por um robô é fazer o robô alcançar e seguir um caminho pré determinado sem restrições de tempo, ou seja, com uma velocidade livremente selecionada. O caminho é definido por uma função paramétrica  $c(s) = (x(s), y(s)) \in \mathbb{R}^2$ , onde  $s$  é chamada de abscissa curvilínea. Note-se que aqui, embora estejamos lidando com uma formação de dois robôs, o robô virtual que caracteriza a formação, que coincide com o VTNT, é quem deve seguir o caminho, enquanto o VANT navega para manter a formação desejada com o VTNT. Por esta razão é que o caminho é definido apenas no plano. Note-se, também, que uma vez definido o caminho, o mesmo é estático em relação ao tempo. Entretanto, o sistema de controle deve inicialmente determinar o ponto do caminho mais próximo da posição atual do robô virtual, assim como a distância entre esses pontos, já que a posição do robô em relação ao caminho depende do tempo, dado que o robô está se movendo para buscar/seguir o caminho.

O ponto do caminho mais próximo do robô virtual é definido como  $c_n = [c_{n_x} \ c_{n_y}]^T$ . A distância entre  $c_n$  e  $[x_f \ y_f]^T$  é definida como  $\tilde{n}$ ,  $\mathbf{t}$  é o vetor unitário tangente ao caminho no ponto  $c_n$  e, por fim,  $\psi_t$  é a orientação de  $\mathbf{t}$  no sistema global de referência. A Figura 5 ilustra essas definições.

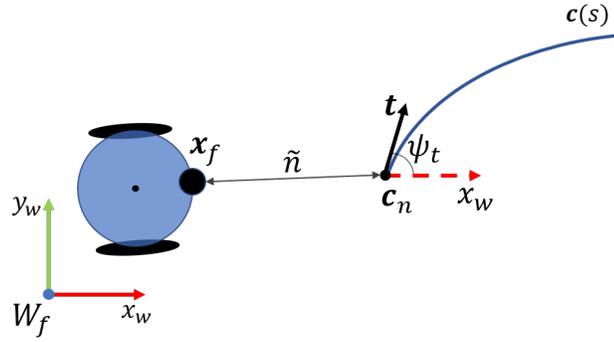


Figura 5 – Exemplo das variáveis de caminho.

Em uma tarefa de seguimento de trajetória, a velocidade desejada seria calculada através da derivada temporal da posição desejada, a qual varia com o tempo, nesse caso. Entretanto, para o seguimento de caminhos isto não ocorre, já que ele é estático em relação ao tempo e, logo, sua derivada em relação ao tempo é sempre nula. Desta maneira, a velocidade desejada durante a navegação é um vetor  $\mathbf{V} \in \mathbb{R}^2$ , cujas componentes são as projeções da velocidade  $V_e$ , que é definida na Equação (2.21), sempre tangente ao caminho, nos eixos  $x$  e  $y$  do eixo inercial, garantindo assim que o robô se mantenha sobre o caminho durante todo o percurso, sempre que ele já estiver em algum ponto do mesmo. Em outras palavras,

$$\mathbf{V} = \begin{bmatrix} \dot{x}_{fd} \\ \dot{y}_{fd} \end{bmatrix} = \begin{bmatrix} V_e \cos \psi_t \\ V_e \sin \psi_t \end{bmatrix}, \quad (2.20)$$

onde  $V_e$  é a entrada de referência de velocidade para o controlador, definida como

$$V_e = \begin{cases} 0 & \text{para } \tilde{n} > e_c, \\ V_d & \text{caso contrário,} \end{cases} \quad (2.21)$$

onde  $e_c$  é um valor de limiar para o erro de posição (ele define um corredor estreito de largura  $2e_c$  dentro do qual o robô virtual, e por consequência o VTNT, deve permanecer), e  $V_d$  é a velocidade desejada do robô virtual ao longo do caminho, selecionada livremente.

Por último, na Equação (2.20) omitiu-se o valor de  $z_{fd}$ , já que o robô virtual navega sempre no plano  $x_w y_w$  (ver Figura 5) onde  $z = 0$ . Além disso, as variáveis responsáveis pela forma da formação  $[\rho_{fd} \quad \alpha_{fd} \quad \beta_{fd}]^T$  também foram omitidas, pois elas não influenciam no seguimento de caminho pelo robô virtual.

## 2.4 Controlador de Seguimento de Caminho

Nesta seção será discutido o controlador utilizado para o cumprimento da tarefa de seguimento de caminho. Inicialmente se aborda o controlador principal da formação e de que maneira as ações de controle calculadas no espaço da formação se transformam em comandos para cada robô individualmente. Em seguida, discute-se o controle de orientação para o VANT, além de um bloco de compensação dinâmica para o mesmo, de maneira a melhorar seu desempenho. Após isso, uma estratégia para redução de velocidade em curvas acentuadas ao longo do caminho é apresentada e tem seu uso justificado. Por último, apresentam-se os conceitos de espaço nulo, sua utilidade e de que maneira a técnica se encaixa no controlador desenvolvido neste trabalho.

### 2.4.1 Sistema Multicamadas de Controle

O controlador responsável pelo seguimento de caminho será dividido em camadas, onde cada uma delas será responsável por um aspecto do controle. A primeira delas é a camada de parametrização do caminho. Pelo fato do caminho ser estático, como já discutido, tal parametrização pode ser realizada *offline*, ou seja, não precisa ser repetida a cada iteração do controle. A segunda camada é a responsável pelo controle da formação, onde se encontra a principal lei de controle. A terceira camada é chamada de camada dos robôs, e é onde os comandos gerados pela camada de controle da formação são distribuídos para os dois robôs que constituem a formação. Por último, a camada do ambiente é onde a ação ocorre, a cada iteração do controle, e os seus efeitos, a mudança de velocidade e posição dos robôs, são medidas e realimentadas para a camada de controle, para a próxima iteração. A Figura 6 mostra graficamente o fluxo de informações entre as camadas.

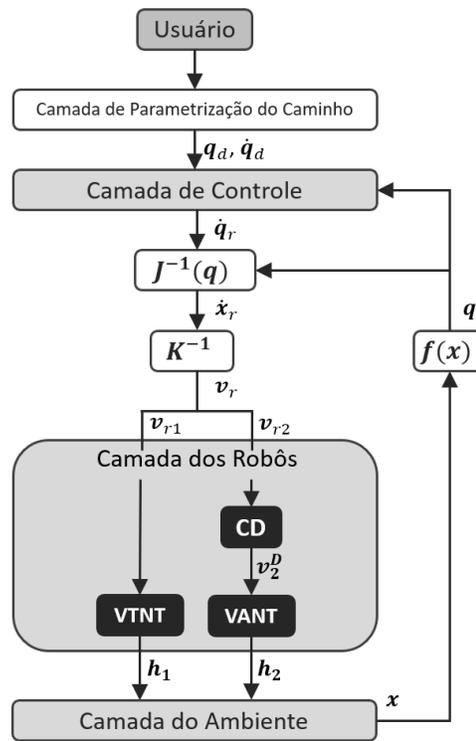


Figura 6 – Estrutura do controlador responsável por guiar a formação VTNT-VANT no seguimento de um caminho especificado.

A lei de controle da formação a ser implementada na camada de controle é uma adaptação da lei de controle adotada em (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020), onde se implementava uma tarefa de seguimento de trajetória. Assim, a nova lei de controle se apresenta como

$$\dot{\mathbf{q}}_r = \mathbf{V} + \mathbf{L}_1 \tanh(\mathbf{L}_2 \tilde{\mathbf{q}}), \quad (2.22)$$

onde  $\dot{\mathbf{q}}_r$  são as referências de velocidade, no espaço da formação, que o sistema deve tomar para que a tarefa de controle seja cumprida,  $\mathbf{L}_1$  e  $\mathbf{L}_2$  são matrizes definidas positivas de saturação e ganho, respectivamente, e  $\tilde{\mathbf{q}}$  são os erros instantâneos de formação, definidos como

$$\tilde{\mathbf{q}} = \mathbf{q}_d - \mathbf{q}, \quad (2.23)$$

sendo  $\mathbf{q}_d$  o vetor de valores desejados para as variáveis da formação, caracterizado como

$$\mathbf{q}_d = \begin{bmatrix} x_{fd} & y_{fd} & z_{fd} & \rho_{fd} & \alpha_{fd} & \beta_{fd} \end{bmatrix}^T. \quad (2.24)$$

O sistema de controle de malha fechada é assintoticamente estável, o que pode ser provado através da teoria de Lyapunov, de maneira análoga à prova apresentada em (RABELO; BRANDÃO; SARCINELLI-FILHO, 2020).

A maneira que o sistema de controle age pode ser assim descrita: sempre que o robô virtual estiver fora do caminho definido, a distância entre o robô e o ponto mais próximo do caminho é considerada. Se tal distância for maior que o limiar  $e_c$ ,  $V_e$  é configurada como zero (ver Eq. (2.21)), o que resulta em  $\dot{x}_{fd} = \dot{y}_{fd} = 0$ . Desta maneira, a tarefa de seguimento de caminho é alterada para uma tarefa de posicionamento, onde o robô se encaminha para o ponto mais próximo do caminho. Conforme ele se aproxima de  $c_n$ , o erro de distância  $\tilde{n}$  vai diminuindo, até que seja menor que  $e_c$ , entrando assim no segundo caso da Eq. (2.21), para o qual  $V_e = V_d$ . O robô virtual irá então desenvolver a velocidade  $V_d$ , ainda que apresente um erro de posição não nulo. Isto significa que ainda existe um valor residual no segundo termo da lei de controle na Eq. (2.22). Isto garante que o robô virtual irá convergir para  $c_n$  de maneira suave, enquanto segue o caminho. Por fim, o erro de posição do robô virtual será igual ou menor que  $e_c$ , em magnitude, o que significa que a formação está seguindo o caminho com a velocidade desejada  $V_d$  com um erro menor que o máximo admitido.

Continuando a análise da Figura 6, após a determinação dos sinais de referência no espaço da formação na Eq. (2.22), ainda é necessário levá-los para o espaço dos robôs. Isto é feito através da derivada no tempo da Eq. (2.18), dada por

$$\dot{\mathbf{x}}_r = \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{q}}_r, \quad (2.25)$$

o que resulta em  $\dot{\mathbf{x}}_r = \begin{bmatrix} \dot{x}_{1r} & \dot{y}_{1r} & \dot{z}_{1r} & \dot{x}_{2r} & \dot{y}_{2r} & \dot{z}_{2r} \end{bmatrix}^T$ , que são as velocidades comandadas agora traduzidas para o espaço dos robôs, mas ainda no sistema global de referência. Quanto ao termo  $\mathbf{J}^{-1}(\mathbf{q})$ , ele é a matriz Jacobiana associada à transformação inversa na Eq. (2.18). Logo,

$$\mathbf{J}^{-1}(\mathbf{q}) = \frac{(\partial \mathbf{x})}{(\partial \mathbf{q})}, \quad (2.26)$$

de tal maneira que

$$\mathbf{J}^{-1}(\mathbf{q}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & c_{\alpha_f} c_{\beta_f} & -\rho_f s_{\alpha_f} c_{\beta_f} & -\rho_f c_{\alpha_f} s_{\beta_f} \\ 0 & 1 & 0 & s_{\alpha_f} c_{\beta_f} & \rho_f c_{\alpha_f} c_{\beta_f} & -\rho_f s_{\alpha_f} s_{\beta_f} \\ 0 & 0 & 1 & s_{\beta_f} & 0 & \rho_f c_{\beta_f} \end{bmatrix}, \quad (2.27)$$

onde  $c$  e  $s$  representam as funções cosseno e seno, respectivamente.

Entretanto, os comandos devem ser enviados ao robô em seu próprio sistema de referência, o que é feito através de

$$\mathbf{v}_r = \mathbf{K}^{-1}\dot{\mathbf{x}}_r, \quad (2.28)$$

onde  $\mathbf{K}^{-1}$  é uma matriz bloco diagonal para a qual cada bloco é a matriz de cinemática inversa de cada um dos robôs que compõem a formação. Desta forma,  $\mathbf{K}^{-1}$  é definida como

$$\mathbf{K}^{-1} = \begin{bmatrix} \mathbf{K}_p^{-1} & 0 \\ 0 & \mathbf{K}_b^{-1} \end{bmatrix} = \begin{bmatrix} c\psi_1 & s\psi_1 & 0 & 0 & 0 & 0 \\ \frac{-s\psi_1}{a} & \frac{c\psi_1}{a} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c\psi_2 & s\psi_2 & 0 \\ 0 & 0 & 0 & -s\psi_2 & c\psi_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.29)$$

Observe-se que foi acrescentada uma dimensão na matriz de cinemática inversa  $\mathbf{K}_p^{-1}$  do *Pioneer 3-DX*. Isto ocorre devido à consideração de um valor de  $z$  para a posição do VTNT, mesmo que ela sempre seja nula. Já em relação ao vetor  $\mathbf{v}_r$ , ele é a referência de velocidade gerada pelo controlador de formação no referencial dos próprios robôs, de tal maneira que

$$\mathbf{v}_r = \begin{bmatrix} \mathbf{v}_{r_1} \\ \mathbf{v}_{r_2} \end{bmatrix}, \quad (2.30)$$

onde  $\mathbf{v}_{r_1}$  é um vetor que contém as velocidades de referência (linear e angular) para o primeiro robô (VTNT) em seu próprio referencial, e  $\mathbf{v}_{r_2}$  representa o mesmo para o segundo robô (VANT) (velocidades lineares nos três eixos) no seu próprio referencial. Em relação à velocidade angular do VANT esta será tratada na Subseção 2.4.2.

### 2.4.2 Controle de Orientação do VANT

O mapeamento do espaço dos robôs para o espaço da formação, mostrado na Eq. (2.16), tem como argumentos apenas valores referentes às posições dos robôs e, naturalmente, todas as considerações feitas até agora se restringem a elas. Devido à restrição não holonômica do VTNT adotado, este sempre estará orientado na mesma direção que seu movimento, ou seja, sua orientação é puramente dependente do caminho. O VANT, entretanto, tem movimento no espaço cartesiano completamente livre, dada sua característica omnidirecional. Assim sendo, sua orientação pode ser desacoplada do seu posicionamento.

Dependendo da tarefa a orientação do VANT ( $\psi_2$ ) pode não ser relevante, podendo até ser ignorada. Entretanto, ela pode ser importante em alguma situação. Por exemplo, caso se deseje utilizar a capacidade de visão do veículo é interessante poder apontá-lo numa direção específica, o que exige orientar o veículo convenientemente. Pode-se, por exemplo, orientar o VANT na mesma direção que o VTNT ( $\psi_2 = \psi_1$ ) para detecção de obstáculos ou outros perigos, por exemplo.

Por este motivo, um controlador de orientação foi inserido no sistema, definido como

$$\dot{\psi}_r = \dot{\psi}_d + L_3 \tanh(L_4 \tilde{\psi}) \quad (2.31)$$

onde  $L_3$  e  $L_4$  são valores de ganho e saturação, respectivamente,  $\dot{\psi}_d$  é a velocidade rotacional desejada em torno do eixo  $z$ , e  $\tilde{\psi}$  é o erro de orientação. Este controlador é similar àquele da Eq. (2.22), exceto que esta versão lida apenas com uma dimensão, a orientação do veículo aéreo.

Como o foco deste trabalho foi em aplicações onde o VANT aponta na mesma direção que o VTNT, a velocidade rotacional desejada e o erro de orientação foram definidos como

$$\dot{\psi}_d = \dot{\psi}_1 = \omega_1 \quad (2.32)$$

e

$$\tilde{\psi} = \psi_d - \psi_2, \quad (2.33)$$

onde

$$\psi_d = \psi_1. \quad (2.34)$$

Este controlador faz com que o VANT busque e mantenha a mesma orientação que o VTNT, esta última sendo dependente do caminho. Portanto, ambos os veículos se orientarão de acordo com a tangente ao caminho.

Na Subseção 2.4.1  $\mathbf{v}_{r_2}$  foi definido como um vetor representando as velocidades cartesianas de referência para o VANT ao longo dos três eixos. Agora, a velocidade rotacional de referência  $v_{\psi_{r_2}}$ , calculada nesta Subseção é anexada a este vetor, e  $\mathbf{v}_{r_2}$  é redefinido como

$$\mathbf{v}_{r_2} = \begin{bmatrix} v_{x_{r_2}} \\ v_{y_{r_2}} \\ v_{z_{r_2}} \\ v_{\psi_{r_2}} \end{bmatrix}. \quad (2.35)$$

### 2.4.3 Compensador Dinâmico

Até agora o controlador foi desenvolvido considerando um robô virtual sem massa, de forma que apenas efeitos cinemáticos foram levados em consideração. Experimentos iniciais mostraram que o VTNT apresentou bons resultados de navegação utilizando apenas o controlador cinemático, diferentemente do VANT, cujo desempenho não foi bom. Então, para melhorar o desempenho do VANT um módulo de compensação dinâmica foi adicionado (bloco CD na Figura 6) de maneira a compensar os efeitos dinâmicos associados ao robô.

O modelo usado para projetar o compensador é o descrito na Eq. (2.8) e o compensador em si, baseado em (PINTO et al., 2020), é então definido como

$$\mathbf{v}_2^D = \mathbf{K}_u^{-1}(\dot{\mathbf{v}}_{r_2} + \mathbf{L}_d(\mathbf{v}_{r_2} - \mathbf{v}_2) + \mathbf{K}_v \mathbf{v}_2), \quad (2.36)$$

onde  $\mathbf{v}_2^D$  é a saída do compensador dinâmico,  $\mathbf{K}_u = \text{diag}([K_1 \ K_3 \ K_5 \ K_7])$  e  $\mathbf{K}_v = \text{diag}([K_2 \ K_4 \ K_6 \ K_8])$  são matrizes diagonais com os parâmetros do modelo,  $\mathbf{L}_d$  é uma matriz diagonal definida positiva de ganhos,  $\dot{\mathbf{v}}_{r_2}$  é a derivada em relação ao tempo de  $\mathbf{v}_{r_2}$ , e  $\mathbf{v}_2$  é o vetor de velocidades do veículo em seu próprio referencial.

Tal compensador muda a dinâmica do erro de velocidade para uma dinâmica linear assintoticamente estável (técnica de linearização por realimentação (KHALIL, 2002)), garantindo que o erro entre as velocidades  $\mathbf{v}_{r_2}$  (velocidade desejada) e  $\mathbf{v}_2$  (velocidade efetiva) convirja a zero assintoticamente.

### 2.4.4 Estratégia de Redução de Velocidade

Por fim, como a velocidade  $V_d$  pode ser selecionada livremente, independente da configuração do caminho, algumas considerações devem ser feitas em relação à sua viabilidade. Por exemplo, uma velocidade compatível com uma estrada reta ou uma curva suave pode ser excessiva em uma curva acentuada. Desta maneira uma estratégia de redução de velocidade, como proposta em (BACHETI; BRANDÃO; SARCINELLI-FILHO, 2020), foi implementada, consistindo na redução da velocidade  $V_d$  quando o ângulo da tangente ao caminho estiver acima de um certo limiar. Isto imita o jeito que um ser humano dirige um automóvel, e permite a

configuração de uma velocidade maior  $V_{d_{max}}$  para a tarefa, que será, na realidade, a velocidade máxima ao longo do caminho.

Desta forma, quando o robô virtual entra em uma curva acentuada sua velocidade, e por consequência a velocidade da formação, será automaticamente reduzida para um valor viável, retornando ao valor original somente após completar a curva. Note-se que uma alta velocidade não poderia ser selecionada se esta estratégia de redução nas curvas não estiver sendo usada, pois ela faria com que a formação não conseguisse realizar as curvas, o que pode ser perigoso.

Para implementar tal esquema a velocidade angular do VTNT,  $\dot{\psi}_1$ , é considerada, de maneira que a velocidade desejada do robô virtual é

$$V_d = \begin{cases} \frac{V_{d_{max}}}{1+L_c|\dot{\psi}_1|} & \text{para } |\dot{\psi}_1| > 0.5rad/s, \\ V_{d_{max}} & \text{caso contrário,} \end{cases} \quad (2.37)$$

ao invés de um  $V_d$  constante. A variável  $L_c$  é um ganho definido para ajuste fino do comportamento da estratégia discutida.

Resumindo, após a definição do caminho a ser seguido e do ajuste do controle, a velocidade  $V_{d_{max}}$  é especificada, ao invés de uma velocidade constante  $V_d$ , que corresponde ao valor máximo de velocidade permitida ao longo do caminho. A velocidade efetivamente realizada pelo robô será igual a  $V_{d_{max}}$  em retas e curvas suaves, porém será reduzida para velocidades menores em curvas acentuadas, de maneira a permitir que a formação execute a curva com segurança.

## 2.5 Controlador Baseado em Espaço Nulo

O controle da formação discutido pode ser entendido como uma tarefa única onde os robôs que a executam buscam ao mesmo tempo posicioná-la no mundo e manter a forma desejada para a mesma. Por outro lado, existem situações onde pode ser mais vantajoso priorizar o posicionamento do robô virtual em detrimento da forma da estrutura, ou vice-versa. Desta maneira, busca-se uma maneira de separar uma tarefa única de controle em duas tarefas, além de se garantir uma hierarquia desejada de cumprimento de tais tarefas.

O método escolhido neste trabalho, Controle Baseado em Espaço Nulo, é apresentado em (ANTONELLI; ARRICHELLO; CHIAVERINI, 2008), onde, além de haver uma separação analítica entre as tarefas, o que pretere a necessidade de comutação de controladores, existe a garantia do cumprimento da tarefa de mais alta prioridade. As tarefas de menor prioridade são cumpridas apenas quando não houver conflito com tarefas de maior prioridade. Entretanto, esta técnica requer uma expressão analítica diferenciável para cada tarefa.

As variáveis de formação, como definidas na Eq. (2.15), podem ser reescritas como

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_p^T & \mathbf{q}_s^T \end{bmatrix}^T \quad (2.38)$$

onde

$$\mathbf{q}_p = \begin{bmatrix} x_f & y_f & z_f \end{bmatrix}^T \quad (2.39)$$

representa as variáveis de posição do robô virtual em relação ao mundo, e

$$\mathbf{q}_s = \begin{bmatrix} \rho_f & \alpha_f & \beta_f \end{bmatrix}^T \quad (2.40)$$

representa as variáveis de forma, ou *shape*, da estrutura virtual no espaço 3D.

Esta estratégia permite a separação de  $\mathbf{J}(\mathbf{x})$ , que é a matriz Jacobiana associada à transformação direta vista na Eq. (2.17), em duas, a saber,

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \mathbf{J}_p(\mathbf{x}) \\ \mathbf{J}_s(\mathbf{x}) \end{bmatrix} \quad (2.41)$$

em que  $\mathbf{J}_p(\mathbf{x})$  mapeia o efeito que as velocidades dos dois robôs têm nas velocidades do robô virtual, e  $\mathbf{J}_s(\mathbf{x})$  mapeia o efeito destas mesmas velocidades nas variações temporais da forma da estrutura virtual. Desta maneira, se caracterizam duas tarefas diferentes, com expressões analíticas diferenciáveis.

Na Subseção 2.4.1, onde foi discutido o controlador, apresentou-se a Equação (2.25), responsável por passar as referências do controlador do espaço da formação para o espaço dos robôs. Para a aplicação da técnica de espaço nulo tal equação é modificada, separando-se a matriz Jacobiana original em seus elementos formadores. Daí, enquanto a tarefa de maior prioridade é tratada da mesma forma, apenas usando a submatriz Jacobiana correspondente, a tarefa com menor prioridade é projetada no espaço nulo da submatriz Jacobiana correspondente à tarefa de maior prioridade. Por exemplo, considerando a tarefa de posicionamento do robô virtual como aquela de maior prioridade altera-se a Equação(2.25) para

$$\dot{\mathbf{x}}_r = \mathbf{J}_p^\dagger \dot{\mathbf{q}}_{p_r} + (\mathbf{I} - \mathbf{J}_p^\dagger \mathbf{J}_p) \mathbf{J}_s^\dagger \dot{\mathbf{q}}_{s_r}, \quad (2.42)$$

onde  $\dot{\mathbf{q}}_{p_r}$  e  $\dot{\mathbf{q}}_{s_r}$  são os sinais de controle da formação referentes ao seu posicionamento e forma, respectivamente,  $\mathbf{I}$  é uma matriz identidade de dimensão apropriada, e  $\mathbf{J}_p^\dagger$  e  $\mathbf{J}_s^\dagger$  são as pseudo inversas de  $\mathbf{J}_p$  e  $\mathbf{J}_s$  (tratam-se de matrizes não quadradas).

Por outro lado, caso a tarefa de manutenção da forma da estrutura virtual seja considerada como prioritária, a Eq. (2.25) é alterada para

$$\dot{\mathbf{x}}_r = \mathbf{J}_s^\dagger \dot{\mathbf{q}}_{s_r} + (\mathbf{I} - \mathbf{J}_s^\dagger \mathbf{J}_s) \mathbf{J}_p^\dagger \dot{\mathbf{q}}_{p_r}. \quad (2.43)$$

A Figura 7 mostra a estrutura de controle modificada após a inserção do espaço nulo, no caso da prioridade de forma.

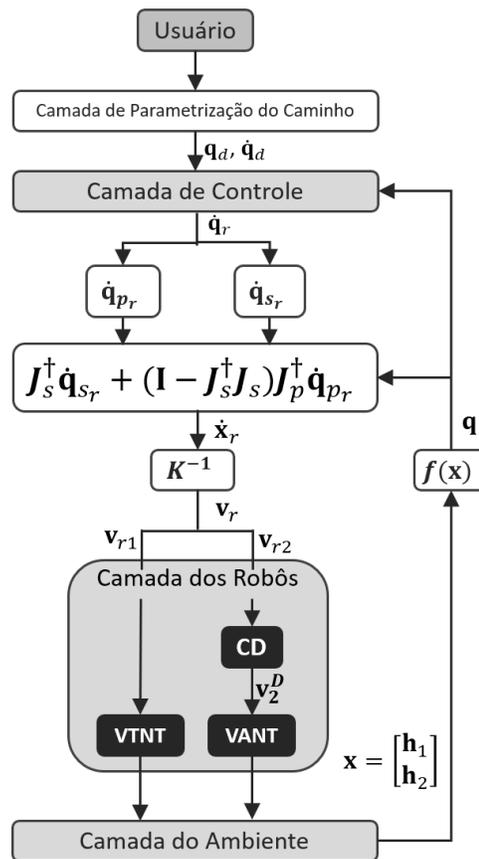


Figura 7 – Estrutura de controle adotada, incluindo o uso da técnica de espaço nulo, dando prioridade maior à manutenção da forma da formação.

## 3 Desvio de Obstáculos

Este capítulo discute o algoritmo desenvolvido para que a formação VTNT-VANT consiga realizar o desvio de obstáculos durante o cumprimento da tarefa de seguimento de caminhos guiada pelo controlador proposto no Capítulo 2. Em primeiro lugar, o desvio de um único obstáculo é abordado, e então o algoritmo é estendido de maneira a lidar com múltiplos obstáculos.

Em uma situação onde existem múltiplos obstáculos que estão distantes entre si o robô virtual, e por consequência a formação, pode voltar ao caminho planejado assim que se desviar de cada obstáculo. Portanto, tal configuração de múltiplos obstáculos pode ser tratada como diversos obstáculos únicos, que portanto não demandam tratamento especial. Desta maneira, o algoritmo para múltiplos obstáculos é usado apenas quando existem obstáculos próximos uns aos outros.

Um importante detalhe a ser considerado é que apenas obstáculos terrestres serão considerados neste trabalho. Portanto, o algoritmo de desvio de obstáculos é associado apenas ao robô virtual que caracteriza a formação, que coincide com o veículo terrestre.

### 3.1 Desvio de um Único Obstáculo

Ao encontrar um obstáculo bloqueando o caminho planejado, o robô virtual deve deixar tal caminho e seguir um caminho alternativo, que seja mais seguro, e que lhe possibilite deixar o obstáculo para trás e retomar o caminho original após isso. Um método que permite que o robô virtual faça tal manobra é proposto, sendo uma solução simples para o problema de desvio em seguimento de caminhos, de fácil interpretação e implementação. O algoritmo emprega geometria simples, de maneira a determinar caminhos seguros em volta do obstáculo, e usa o mesmo controlador de seguimento de caminhos discutido no Capítulo 2 para navegar neste novo caminho. Portanto, a estratégia implementada corresponde à uma estratégia de planejamento de caminhos, não a uma estratégia de controle.

O algoritmo assume que a posição e dimensões de cada obstáculo mais próximo que uma certa distância do robô são conhecidas, após sua detecção pelo robô. A zona de detecção é um círculo centralizado na posição atual  $\mathbf{x}_f$  do robô virtual (posição da formação), e cujo raio é  $r_{det}$ . Adicionalmente, cada obstáculo será modelado como uma circunferência cujo raio é grande o suficiente para envolver todos os limites físicos do obstáculo e mais uma zona de segurança. Tal zona de segurança, por sua vez, é grande o suficiente para permitir que o corpo do robô navegue sem colidir com o obstáculo assim representado. Esta circunferência, definida pelos limites físicos do obstáculo junto com a zona de segurança, é chamada de zona de desvio, e seu raio é  $r_{avd}$ . É suposto que  $r_{det}$  é maior que  $r_{avd}$  para todos os obstáculos (ver Figura 8).

Assim que o robô detecta um obstáculo, dois pontos especiais no caminho original ( $\mathbf{c}_o(s)$ ) que ele está seguindo são definidos. O primeiro é a primeira interseção do caminho  $\mathbf{c}_o(s)$  com a circunferência de raio  $r_{avd}$  centrada na posição do obstáculo  $\mathbf{x}_{o_i}$  (considerando o  $i$ -ésimo obstáculo). O segundo ponto é a segunda interseção do caminho com a circunferência que caracteriza o obstáculo, como mostra a Figura 8. Perceba-se que isso significa que o caminho sendo seguido passa dentro da zona de desvio do obstáculo, o que demanda que uma manobra seja feita para que o robô desvie dele.

Os dois pontos caracterizados acima serão chamados de  $c_{avd_{f_i}}$  e  $c_{avd_{l_i}}$ , respectivamente, e são pontos chave para a decisão se o desvio deve ser realizado ou não, e qual caminho deve ser tomado, caso o desvio deva ser feito. Se tais pontos não existirem significa que o obstáculo está próximo o suficiente para ser detectado, mas a

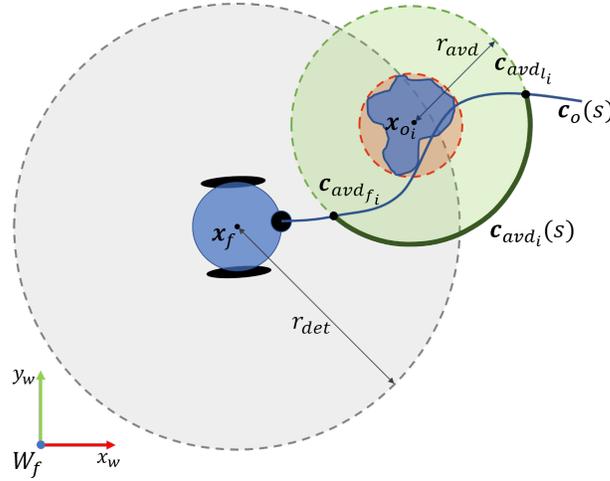


Figura 8 – Caracterização do desvio de obstáculos, considerando um único obstáculo.

circunferência de raio  $r_{avd}$  não toca o caminho. Neste caso o obstáculo não é considerado uma ameaça para a navegação, e nenhuma manobra de desvio é feita. Por outro lado, quando o caminho planejado tem interseção com a circunferência de raio  $r_{avd}$ ,  $c_{avd_{f_i}}$  e  $c_{avd_{i_i}}$  são obtidos, e a estratégia adotada é sair do caminho original e tomar o caminho correspondente a um arco da circunferência que caracteriza o obstáculo, iniciando no ponto  $c_{avd_{f_i}}$  e terminando no ponto  $c_{avd_{i_i}}$ . Assim que o robô atinge o ponto  $c_{avd_{i_i}}$ , ele retoma o caminho original e continua sua navegação. Um detalhe importante é que, como ambos os pontos  $c_{avd_{f_i}}$  e  $c_{avd_{i_i}}$  pertencem a  $\mathbf{c}_o(s)$ , a continuidade entre os caminhos é garantida, ou seja, o robô vai alcançar o novo caminho determinado ao seguir o caminho original, e vai retornar para  $\mathbf{c}_o(s)$  apenas depois de deixar o obstáculo para trás.

Assim que chega em  $c_{avd_{f_i}}$  o robô pode escolher ir para a esquerda ou para a direita, para desviar do obstáculo. Esta escolha poderia ser aleatória ou seguir outro critério, como a direção para a qual o robô está virado ou o caminho mais curto, por exemplo. Neste trabalho o critério do caminho mais curto foi adotado, e o novo caminho  $\mathbf{c}_{avd_i}(s)$ , quando se considera o exemplo ilustrado na Figura 8, é aquele assinalado.

Mesmo que à primeira vista pareça um problema simples determinar  $\mathbf{c}_{avd_i}(s)$  como uma função de arco de circunferência, isto na verdade pode ser complexo, porque  $c_{avd_{f_i}}$ ,  $\mathbf{x}_{o_i}$  e  $c_{avd_{i_i}}$  podem estar em qualquer posição e quadrante no sistema de referência do mundo  $W_f$ . Isto leva a uma complicada análise de quadrante para determinar a abscissa curvilínea para o arco de circunferência. Uma outra complicação a mais é a descontinuidade presente na região de  $[-\pi, \pi]$ .

Para sanar este problema, um novo sistema de referência  $O_f$  é definido, cuja origem é o centro da circunferência que descreve o obstáculo ( $\mathbf{x}_{o_i}$ ) e cujo eixo  $x_0$  está na mesma direção da linha que conecta  $c_{avd_{f_i}}$  e  $\mathbf{x}_{o_i}$ . Isso é feito através da matriz de transformação homogênea  $H_o^w$ . Desta maneira,  $H_o^w$  representa a posição e orientação do novo sistema  $O_f$  com respeito ao sistema global de referência  $W_f$ , e é definida como

$$H_o^w = \begin{bmatrix} \cos \psi_o & -\text{sen} \psi_o & 0 & x_{o_i} \\ \text{sen} \psi_o & \cos \psi_o & 0 & y_{o_i} \\ 0 & 0 & 1 & z_{o_i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

onde  $\mathbf{x}_{o_i} = [x_{o_i} \ y_{o_i} \ z_{o_i}]^T$ ,  $c_{avd_{f_i}} = [x_{c_{avd_{f_i}}} \ y_{c_{avd_{f_i}}} \ z_{c_{avd_{f_i}}}]^T$  e  $\psi_o$  é o ângulo que  $\mathbf{x}_{o_i}$  faz com relação a

$c_{avd_{f_i}}$ , definido como

$$\psi_o = \arctan \frac{y_{o_i} - y_{c_{avd_{f_i}}}}{x_{o_i} - x_{c_{avd_{f_i}}}}. \quad (3.2)$$

Entretanto, para transformar pontos que estão representados no sistema de referência do mundo  $W_f$  para os pontos correspondentes no sistema do obstáculo  $O_f$ , a transformada inversa  $H_w^o$  é necessária. Felizmente, devido às propriedades de transformação homogênea, é possível afirmar que

$$H_w^o = (H_w^w)^{-1}, \quad (3.3)$$

e então

$$\begin{bmatrix} \mathbf{x}_{o_i}^o \\ 1 \end{bmatrix} = H_w^o \begin{bmatrix} \mathbf{x}_{o_i} \\ 1 \end{bmatrix}, \quad (3.4)$$

onde  $\mathbf{x}_{o_i}^o$  é o ponto correspondente à  $\mathbf{x}_{o_i}$  no novo sistema de referência e

$$\begin{bmatrix} c_{avd_{f_i}}^o \\ 1 \end{bmatrix} = H_w^o \begin{bmatrix} c_{avd_{f_i}} \\ 1 \end{bmatrix}, \quad (3.5)$$

assim como

$$\begin{bmatrix} c_{avd_{i_i}}^o \\ 1 \end{bmatrix} = H_w^o \begin{bmatrix} c_{avd_{i_i}} \\ 1 \end{bmatrix} \quad (3.6)$$

onde  $c_{avd_{f_i}}^o$  e  $c_{avd_{i_i}}^o$  são os pontos no novo sistema de referência que correspondem a  $c_{avd_{f_i}}$  e  $c_{avd_{i_i}}$ , respectivamente.

Devido a esta escolha de eixos, e às características intrínsecas dos pontos envolvidos, em toda as situações é possível afirmar que

$$\mathbf{x}_{o_i}^o = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{ e } c_{avd_{f_i}}^o = \begin{bmatrix} -r_{avd} \\ 0 \\ 0 \end{bmatrix}. \quad (3.7)$$

Agora o problema é bastante simplificado, já que dois dos pontos envolvidos são constantes, e apenas o último,  $c_{avd_{i_i}}^o$ , apresenta qualquer variação. Além disso, o ângulo que  $c_{avd_{f_i}}^o$  faz com  $\mathbf{x}_{o_i}^o$ , dado por  $\psi_f^o$ , está exatamente acima da descontinuidade, o que significa que a função do arco de circunferência nunca passará por ela, mas sempre começará a partir dela. Qual valor  $\psi_f^o$  vai assumir,  $\pi$  ou  $-\pi$ , depende na direção desejada para o arco, o que será explorado mais à frente.

A Figura 9 mostra a mesma situação apresentada na Figura 8, porém rotacionada de maneira a deixar o sistema de referência do obstáculo,  $O_f$ , em uma configuração familiar, dando apoio às afirmações feitas nos parágrafos anteriores. Ainda mais,  $\psi_l^o$  é definido como o ângulo de  $c_{avd_{i_i}}^o$  em relação a  $\mathbf{x}_{o_i}^o$ .

Quando estiver desviando de um obstáculo, o robô sempre vai navegar ao longo do arco de circunferência que começa em  $c_{avd_{f_i}}^o$  e termina em  $c_{avd_{i_i}}^o$ . O argumento da função do arco deve começar em  $\psi_f^o$  e terminar em  $\psi_l^o$ . Desta maneira, a abscissa curvilínea é definida como

$$s = \psi_f^o \xrightarrow{inc} \psi_l^o, \quad (3.8)$$

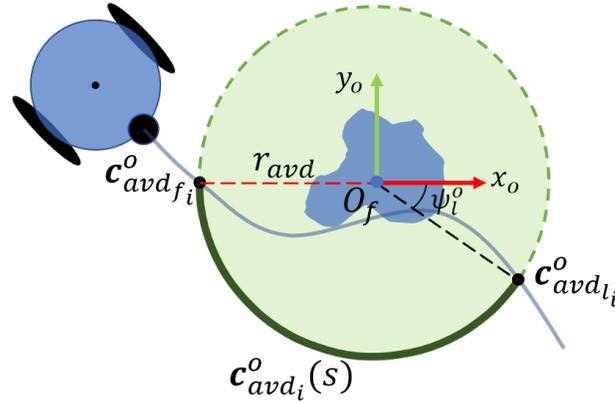


Figura 9 – Situação de desvio de obstáculos considerando o novo sistema de referência, cuja origem é  $O_f$ .

onde

$$\psi_f^o = \begin{cases} -\pi & \text{para } \psi_l^o < 0, \\ \pi & \text{caso contrário,} \end{cases} \quad (3.9)$$

e  $inc$  é o incremento da função e depende da resolução escolhida para o caminho. Essa condição garante que se  $\psi_l^o$  é negativo, como na Figura 9, o robô vai seguir pelo arco de baixo, e se o ângulo for positivo o robô vai seguir pelo arco de cima, portanto sempre seguindo o menor caminho. A partir daí, a função que define o menor arco de circunferência é

$$\mathbf{c}_{avd_i}^o(s) = \begin{bmatrix} r_{avd} \cos s \\ r_{avd} \sin s \\ 0 \end{bmatrix}, \quad (3.10)$$

onde as coordenadas em  $z$  são zero, pois a função é definida no plano  $xy$ , que é onde o robô virtual navega.

Tal função, entretanto, está definida em relação ao sistema  $O_f$ . Para estabelecer a função nas coordenadas globais, onde a navegação acontece, a transformada homogênea inversa é aplicada, o que resulta em

$$\begin{bmatrix} \mathbf{c}_{avd_i}(s) \\ \mathbf{1} \end{bmatrix} = H_o^w \begin{bmatrix} \mathbf{c}_{avd_i}^o(s) \\ \mathbf{1} \end{bmatrix}. \quad (3.11)$$

Portanto, quando o robô detecta um obstáculo durante o seguimento do caminho original, as interseções com o caminho e a zona de desvio do obstáculo são determinadas. Um novo caminho de desvio será então calculado pelo algoritmo, começando no caminho original e seguindo ao longo dos limites da zona de desvio, no caminho mais curto disponível. Quando o caminho de desvio acaba, ele está conectado com o caminho original, de tal maneira que quando o robô atinge este ponto o obstáculo ficou para trás, e o robô retoma sua navegação ao longo do caminho original.

### 3.2 Desvio de Múltiplos Obstáculos

O algoritmo descrito na última seção cobre todos os casos de obstáculos singulares que o robô possa encontrar durante sua navegação por um caminho definido no mundo real. Entretanto, o que aconteceria caso o robô encontrasse um novo obstáculo durante sua navegação pelo caminho temporário gerado pelo algoritmo de desvio? Na seção anterior um dos parâmetros usados era o primeiro ponto de interseção com o caminho

original,  $c_{avd_{f_i}}$ , o que garantia a continuidade com o caminho planejado. Agora, porém, o robô não está mais no caminho original e não faz mais sentido definir  $c_{avd_{f_i}}$ .

Assim é que uma versão modificada do algoritmo desenvolvido na última seção é aplicada quando o robô encontra um novo obstáculo durante a realização de uma manobra de evasão de um obstáculo previamente detectado. Perceba-se que esta é uma possibilidade real quando existem múltiplos obstáculos próximos uns dos outros no ambiente de navegação.

Assume-se que o robô está diante de um novo obstáculo  $\mathbf{x}_{o_{i+1}}$  quando a distância de sua atual posição  $\mathbf{x}_f$  para o centro  $\mathbf{x}_{o_{i+1}}$  da circunferência caracterizando o novo obstáculo fica menor que a distância entre  $\mathbf{x}_f$  e  $\mathbf{x}_{o_i}$ , que é o centro do obstáculo atual do qual o robô está se desviando. Nessa situação novos parâmetros serão calculados, e um procedimento similar ao anterior será implementado. A Figura 10 ilustra a situação e os parâmetros relevantes, considerando dois obstáculos próximos. Como mencionado anteriormente, se os obstáculos da figura não fossem próximos um do outro, eles seriam tratados como dois obstáculos únicos, um após o outro.

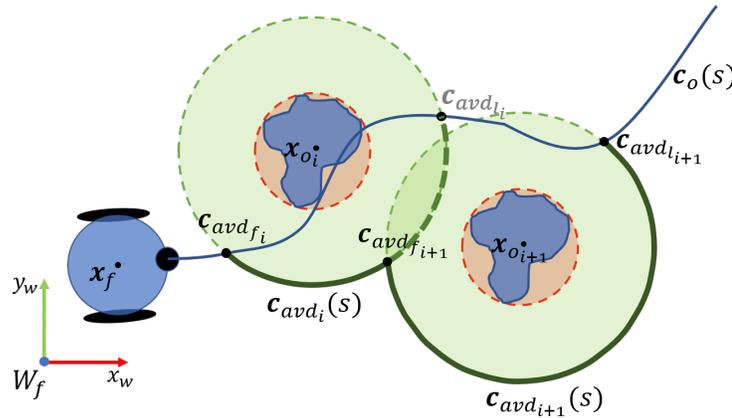


Figura 10 – Desvio de obstáculos considerando múltiplos obstáculos próximos uns dos outros.

A região de detecção que engloba um raio de  $r_{det}$  a partir do centro do robô foi omitida, para clareza na figura. Outro detalhe na situação mostrada é que o segundo obstáculo ainda não terá sido detectado quando o robô começar a manobra de desvio do primeiro. De fato, a detecção do segundo obstáculo ocorrerá após a entrada do robô no caminho criado para desvio do primeiro obstáculo, caracterizado como  $c_{avd_i}(s)$ .

Assim que o robô detecta a presença de um novo obstáculo enquanto desvia de um obstáculo anterior, os pontos  $c_{avd_{f_{i+1}}}$  e  $c_{avd_{i+1}}$  serão determinados, de maneira similar à seção anterior. A diferença é que  $c_{avd_{f_{i+1}}}$  é determinado considerando o caminho temporário que o robô está seguindo atualmente, ou seja  $c_{avd_i}(s)$ , e não sobre o caminho original  $c_o(s)$ . O objetivo, então, é determinar  $c_{avd_{i+1}}(s)$ , que será o novo caminho seguro que o robô seguirá, já que  $c_{avd_i}(s)$  não é mais seguro, devido à presença do novo obstáculo.

A Figura 10 ilustra a situação: o robô segue um caminho seguro para desviar do primeiro obstáculo, quando detecta um segundo obstáculo, e assim ele deve mudar o caminho a ser seguido novamente. Nesta situação se o algoritmo de múltiplos obstáculos usasse o mesmo critério do algoritmo anterior, o caminho mais curto, o arco escolhido levaria o robô de encontro ao obstáculo anterior. Isto significa que o algoritmo não deveria escolher o caminho mais curto. De fato, ele deveria escolher o arco de circunferência mais longo, e desta maneira não seria levado de volta ao obstáculo anterior. Por outro lado, se o segundo obstáculo fosse colocado horizontalmente em cima do primeiro, o robô realmente deveria pegar o caminho mais curto nas duas manobras de desvio. Isto mostra que a escolha do caminho mais seguro não é mais tão simples quanto na

seção anterior. Por conseguinte, um dos caminhos descritos pelos arcos de circunferência possíveis (curto ou longo) sempre vai fazer com que o robô volte ao obstáculo anterior, enquanto o outro será um caminho seguro.

Para solucionar o problema levantado no parágrafo anterior um novo critério é adotado. Após seguir o mesmo procedimento mostrado na seção anterior o algoritmo calcula a função do arco de circunferência mais curto, agora identificado por  $c_{avd_{i+1}^s}(s)$ , com o sobrescrito  $s$  identificando o arco mais curto ( $s$  do inglês *short*). Para calcular o arco mais longo a análise é a mesma, com a diferença de que agora a abcissa curvilínea será o complemento da anterior. Isto significa que para o cálculo do arco longo o procedimento na Eq. (3.8) é o mesmo, mas a Eq. (3.9) é modificada para

$$\psi_f^o = \begin{cases} \pi & \text{para } \psi_l^o < 0, \\ -\pi & \text{caso contrário,} \end{cases} \quad (3.12)$$

o que garante que  $c_{avd_{i+1}^l}(s)$  descreve o arco mais longo, definido como

$$c_{avd_{i+1}^l}^o(s) = \begin{bmatrix} r_{avd} \cos s \\ r_{avd} \sin s \\ 0 \end{bmatrix}, \quad (3.13)$$

e

$$\begin{bmatrix} c_{avd_{i+1}^l}(s) \\ \mathbf{1} \end{bmatrix} = H_w^o \begin{bmatrix} c_{avd_{i+1}^l}^o(s) \\ \mathbf{1} \end{bmatrix}. \quad (3.14)$$

Após a definição de ambos os caminhos, curto e longo, o algoritmo irá calcular a distância de cada um de seus pontos para o obstáculo anterior  $\mathbf{x}_{o_i}$ , e a distância mínima de cada caminho será identificada como  $d_{min_s}$  e  $d_{min_l}$ , para os caminhos curto e longo, respectivamente. Assim, o novo caminho seguro será definido como

$$c_{avd_{i+1}}(s) = \begin{cases} c_{avd_{i+1}^s}(s) & \text{para } d_{min_s} > d_{min_l}, \\ c_{avd_{i+1}^l}(s) & \text{caso contrário.} \end{cases} \quad (3.15)$$

Em outras palavras, o caminho escolhido será aquele mais distante do obstáculo anterior.

Finalmente, ainda que esta discussão tenha se concentrado em dois obstáculos, sempre que o robô encontrar um novo obstáculo em seu caminho quando ainda estiver desviando de um obstáculo prévio, este algoritmo pode ser aplicado novamente. A diferença é que agora levará em conta  $\mathbf{x}_{o_{i+1}}$  e  $\mathbf{x}_{o_{i+2}}$ , até  $n$  obstáculos. Situações envolvendo mais de dois obstáculos serão apresentadas nas simulações e experimentos discutidos mais adiante, e deixarão mais claro o funcionamento do algoritmo.

## 4 Resultados e Discussões

Neste capítulo são apresentados os resultados obtidos em simulações e experimentos utilizando a teoria desenvolvida nos últimos capítulos. Na primeira seção são apresentados os resultados do controlador de formação aplicado ao seguimento de caminhos. A segunda seção apresenta os resultados deste mesmo controlador, porém agora adicionado de um bloco baseado em espaço nulo para hierarquizar as tarefas. Finalmente, na última seção são apresentados os resultados relacionados ao algoritmo de desvio de obstáculos.

Uma visão geral da instalação experimental montada para validar o sistema de controle é mostrada na Figura 11. Nela estão desenhadas as conexões dos três computadores utilizados, que são um desktop Windows rodando o software do sistema de captura de movimentos *OptiTrack*<sup>1</sup>, um notebook Linux rodando os nós da rede ROS<sup>2</sup>, e um notebook Windows rodando o sistema de controle, escrito em Matlab®. Complementando esta visão geral, a Figura 12 ilustra o VTNT e o VANT, mostrando os marcadores colados a eles para permitir que as câmeras do sistema de captura os rastreiem. Note-se também que o *Pioneer 3-DX* tem uma plataforma sobre sua parte superior, que é utilizada para pouso do VANT.

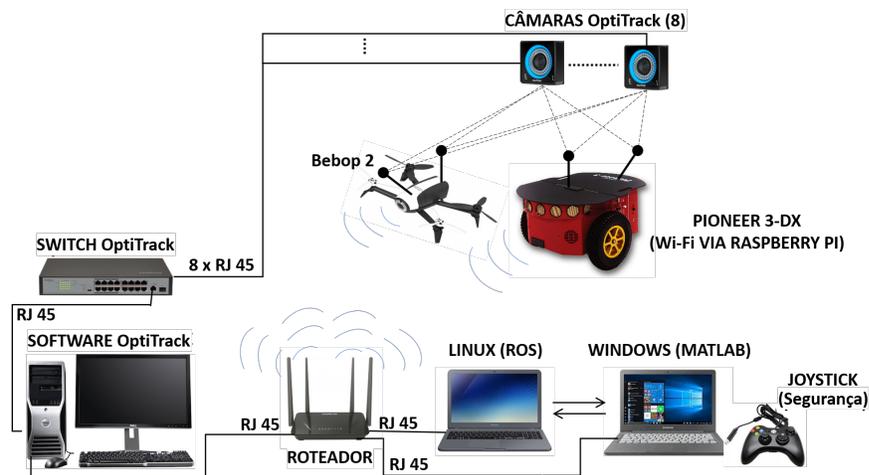


Figura 11 – Uma visão geral do equipamento de hardware usado para rodar o experimento.

Os dados relativos às posições dos robôs foram obtidos usando-se uma configuração com 8 câmeras do *OptiTrack*, que fornece as coordenadas do ponto de interesse para controle em cada um dos veículos, ou seja, os valores de  $\mathbf{h}_1$  e  $\mathbf{h}_2$ . Tal sistema também fornece as orientações dos dois veículos,  $\psi_1$  e  $\psi_2$ . Já as velocidades necessárias são obtidas por diferenciação numérica dos dados de posição e orientação obtidos com o *OptiTrack*.

### 4.1 Seguimento de Caminho e Pouso do VANT sobre o VTNT

Nesta seção são apresentados experimentos do controlador de seguimento de caminho, mostrando-se a navegação da formação e o pouso do VANT sobre o VTNT. Em um primeiro momento são mostrados os experimentos com pouso no VTNT parado, e depois são mostrados os experimentos onde ocorreram pousos

<sup>1</sup> Ver <https://optitrack.com/motion-capture-robotics/>

<sup>2</sup> Ver <https://www.ros.org/>

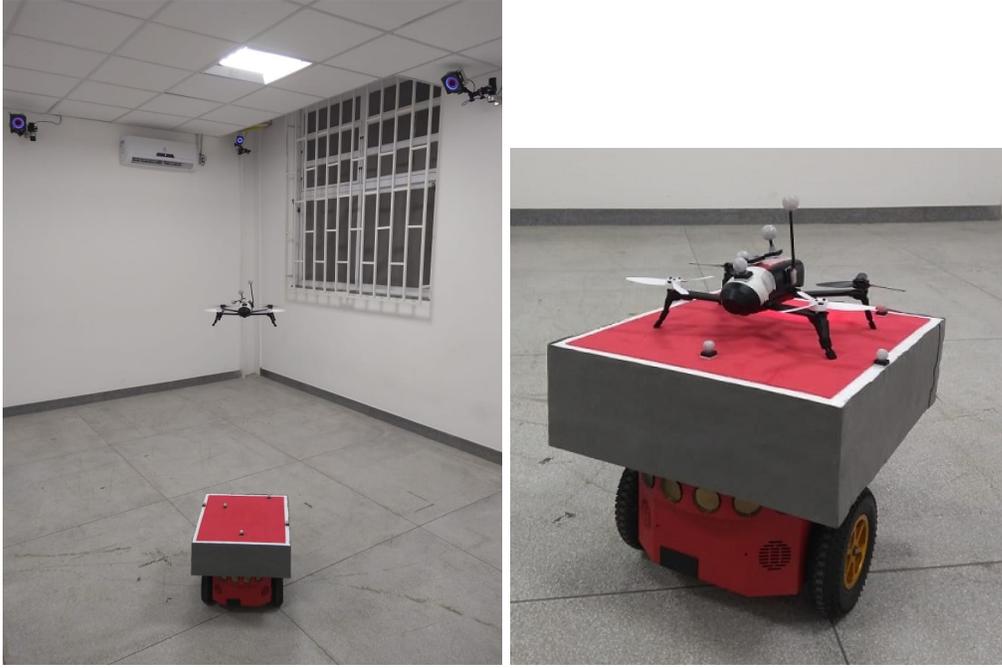


Figura 12 – Foto dos robôs utilizados, assim como do sistema de captura de movimentos *OptiTrack*.

com o VTNT em movimento. Além disso, os primeiros experimentos mostrados não têm a estratégia de redução de velocidade discutida na Seção 2.4.4, o que é proposital, para mostrar a necessidade de tal estratégia.

#### 4.1.1 Pouso Estático

O primeiro experimento apresentado foi conduzido com uma velocidade desejada  $V_d = 0,45 \text{ m/s}$ . Observe-se que a falta do subscrito *max* indica que não há a estratégia de redução de velocidade, como dito anteriormente. O caminho a ser seguido foi definido como

$$\begin{bmatrix} x(s) \\ y(s) \\ z(s) \end{bmatrix} = \begin{bmatrix} 0,8 \text{ sen}\left(\frac{2\pi s}{100}\right) \text{ m} \\ 1,2 \text{ sen}\left(\frac{\pi s}{100}\right) \text{ m} \\ 0 \text{ m} \end{bmatrix}. \quad (4.1)$$

A formação desejada durante a navegação é caracterizada como

$$\mathbf{q}_{d_n} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 1,2 \text{ m} \\ 0 \text{ rad} \\ \frac{\pi}{2} \text{ rad} \end{bmatrix}, \quad (4.2)$$

enquanto que a formação desejada no procedimento do pouso é

$$\mathbf{q}_{d_i} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 0,75 \text{ m} \\ (\psi_1 + \pi) \text{ rad} \\ \frac{4\pi}{9} \text{ rad} \end{bmatrix}, \quad (4.3)$$

que tem o objetivo de centralizar o VANT sobre a plataforma no topo do VTNT (vale lembrar que o ponto de controle do VTNT está deslocado para frente) assim como baixar o VANT a uma altura propícia para que ele caia sobre o VTNT, pois pousar o VANT utilizado significa simplesmente desligar seus motores. Além disso, no momento do pouso  $V_d = 0 \text{ m/s}$ . Finalmente, os ganhos foram ajustados como

$$\begin{aligned} \mathbf{L}_1 &= \text{diag} [1,2 \quad 1,2 \quad 1 \quad 3 \quad 1,5 \quad 1,5], \\ \mathbf{L}_2 &= \text{diag} [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1], \\ L_3 &= 1, \\ L_4 &= 1 \text{ e} \\ \mathbf{L}_d &= \text{diag} [2 \quad 2 \quad 1,8 \quad 5]. \end{aligned} \quad (4.4)$$

A tarefa pedia que a formação navegasse por 30 segundos, e então o VANT deveria pousar no VTNT, que já estará parado. Os resultados estão mostrados nas Figuras 13, 14 e 15. Além disso, um vídeo deste experimento pode ser encontrado no link <<https://youtu.be/IPXWDFqnzSY>>.

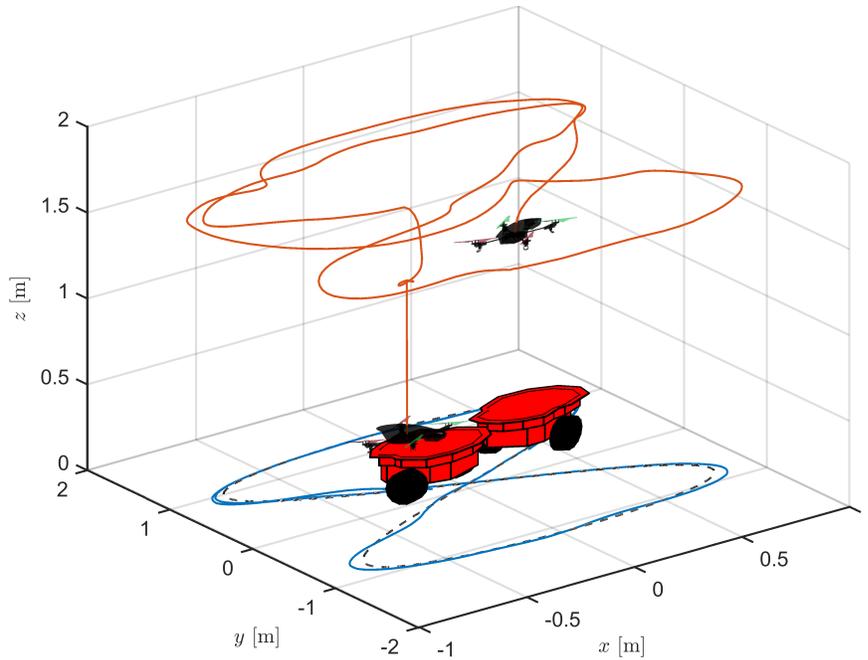


Figura 13 – Caminho seguido pela formação quando  $V_d = 0,45 \text{ m/s}$ .

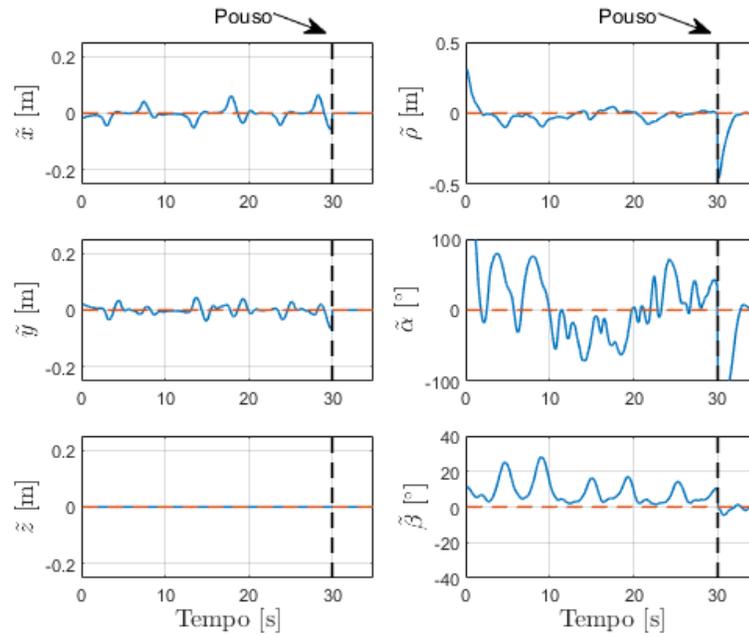


Figura 14 – Erros de formação para a tarefa de seguimento de caminho quando  $V_d = 0,45 \text{ m/s}$ .

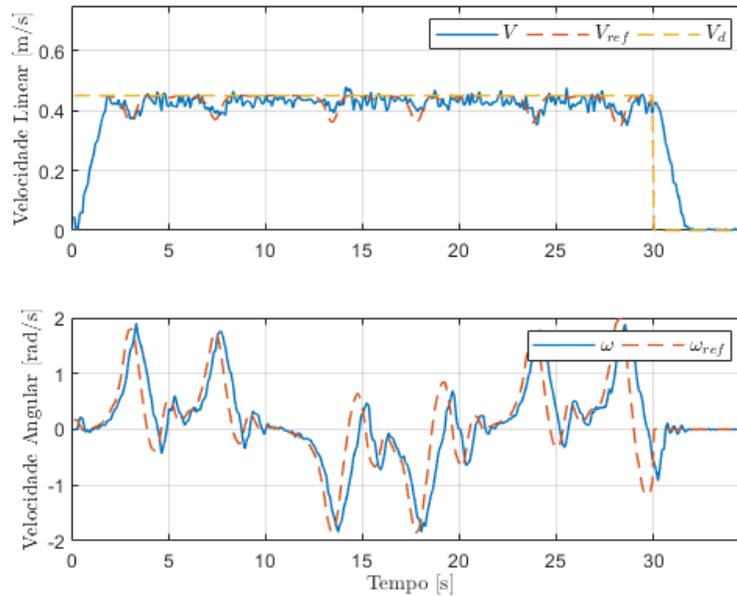


Figura 15 – Velocidades do VTNT para a tarefa de seguimento de caminho quando  $V_d = 0,45 \text{ m/s}$ .

Como pode ser observado nas imagens, a formação foi capaz de seguir o caminho planejado, com erros pequenos. Entretanto, como comentado na Seção 2.4.4, nem todas as velocidades são viáveis em um dado caminho para os robôs. Para tornar isto mais claro, outro experimento foi realizado, nas mesmas condições, porém com uma velocidade desejada  $V_d = 0,5 \text{ m/s}$ . O resultado, mostrado na Figura 16, é que a formação falha no seguimento do caminho, apresentando grandes erros. Um vídeo de tal experimento pode ser visto no link <<https://youtu.be/IPXWDFqnzSY?t=51>>.

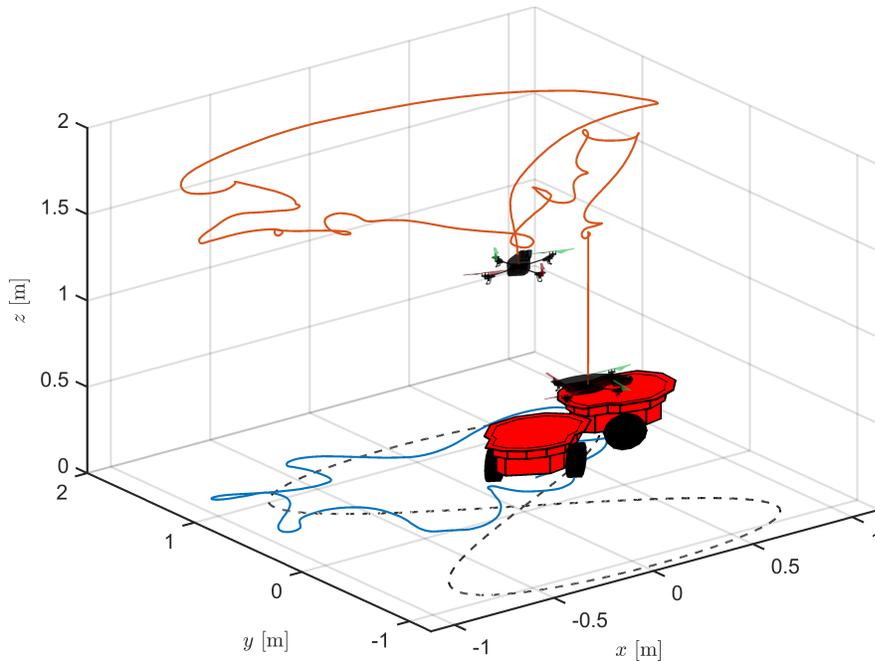


Figura 16 – Caminho seguido pela formação quando  $V_d = 0,5 \text{ m/s}$ .

Este último experimento mostrou a necessidade da redução da velocidade em curvas acentuadas, e a estratégia discutida na Seção 2.4.4 foi introduzida. Após a introdução de tal estratégia o experimento em que  $V_d = 0,45 \text{ m/s}$  foi repetido, agora com  $V_{d_{max}} = 0,45 \text{ m/s}$  foi repetido mantendo-se todos os ganhos e demais condições, para efeito de comparação. O resultado está resumido nas Figuras 17, 18 e 19, onde pode ser visto que os erros de formação no seguimento de caminho são muito menores do que no caso anterior, quando a estratégia de redução da velocidade nas curvas acentuadas não estava sendo utilizada. Além disso, é possível observar que sempre que a velocidade angular do VTNT é alta sua velocidade linear cai, o que se deve à estratégia adotada, pois a velocidade angular do VTNT é um indicativo do quão acentuada é a curva sendo percorrida. O experimento pode ser melhor compreendido ao se assistir o vídeo presente em <https://youtu.be/IPXWDFqnzSY?t=99>.

No final, o mesmo experimento foi repetido novamente, agora com uma velocidade muito maior, a saber  $V_{d_{max}} = 0,65 \text{ m/s}$ , que é bem próxima da velocidade máxima do *Pioneer 3-DX* ( $0,75 \text{ m/s}$ ). A Figura 20 mostra que a tarefa foi cumprida mesmo quando se considera esta alta velocidade. O link <https://youtu.be/IPXWDFqnzSY?t=149> mostra o vídeo deste experimento.

#### 4.1.2 Pouso em Movimento

O próximo experimento mostra uma tarefa de navegação que envolve o VANT decolando de um VTNT estático, navegando por alguns segundos e então pousando em um VTNT em movimento. Após um tempo, o VANT decola novamente, desta vez do VTNT em movimento, e após um intervalo o VANT faz seu pouso final no VTNT estático.

Para cumprir tal tarefa, as formações desejadas  $\mathbf{q}_{d_n}$  e  $\mathbf{q}_{d_t}$  da subseção anterior foram usadas, tendo-se adotada a velocidade  $V_{d_{max}} = 0,50 \text{ m/s}$ , o que corresponde a dois terços da velocidade máxima do VTNT.

A navegação correspondente a tal experimento é mostrada na Figura 21. Ela foi dividida em três partes, para uma apresentação mais clara dos resultados. A primeira parte mostra do início da navegação até o

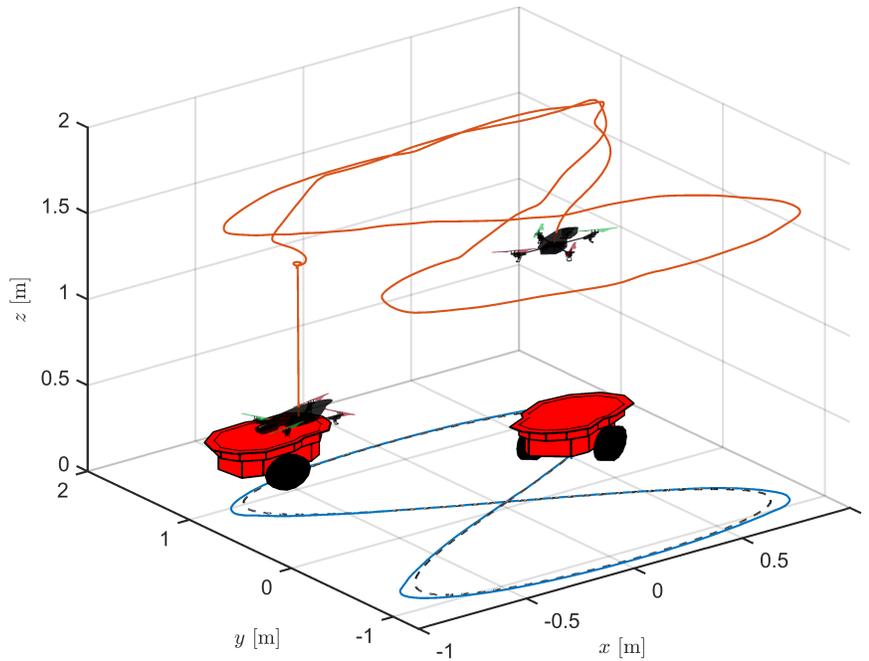


Figura 17 – Caminho percorrido pela formação quando  $V_{d_{max}} = 0,45 \text{ m/s}$ , com a estratégia de mudar  $V_d$  ao longo do caminho.

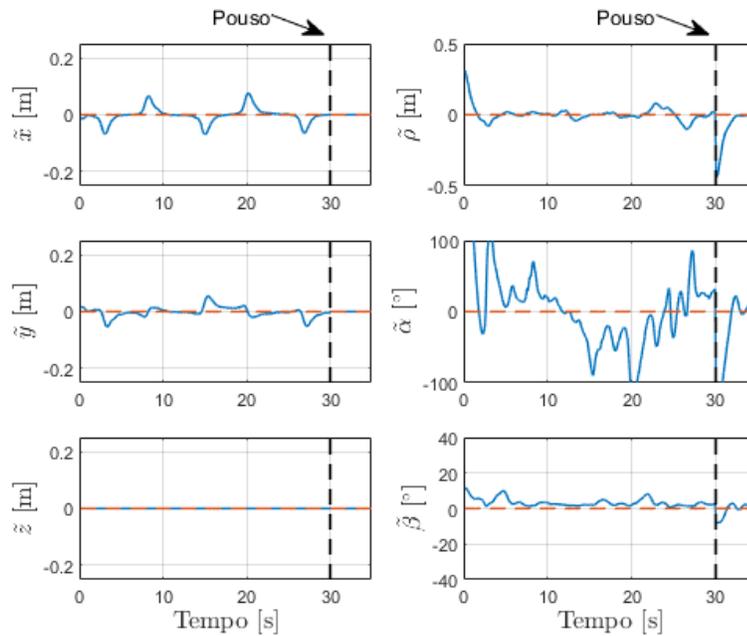


Figura 18 – Erros de formação quando corrigindo a velocidade no caminho,  $V_{d_{max}} = 0,45 \text{ m/s}$ .

momento que o VANT pousa no VTNT em movimento (os ícones correspondentes aos robôs estão representados no início e no final de cada parte do experimento). É possível observar que inicialmente o VANT descreveu um arco mais amplo que o do VTNT, e que rapidamente eles se alinham verticalmente, com o veículo aéreo exatamente sobre o terrestre, como pedem as variáveis de formação desejadas.

A segunda parte começa com o início da navegação onde o VANT está pousado e termina assim que o

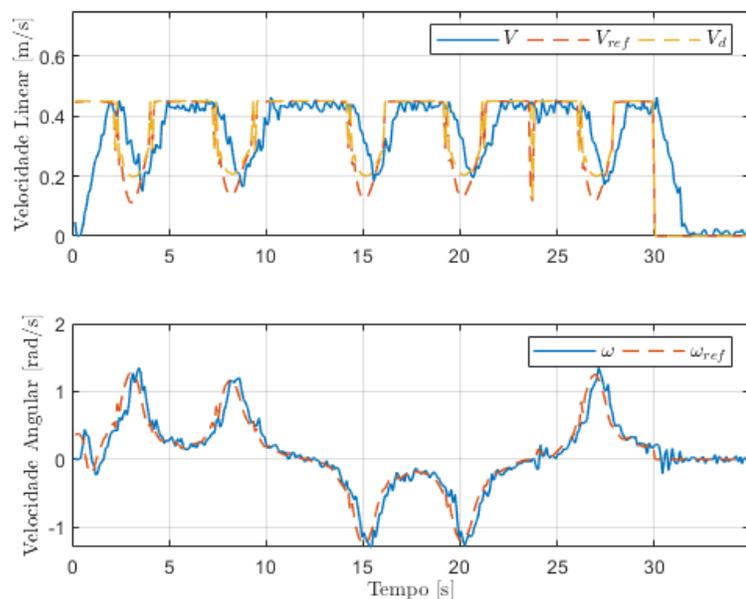


Figura 19 – Velocidades do VTNT quando corrigindo a velocidade no caminho,  $V_{d_{max}} = 0,45 \text{ m/s}$ .

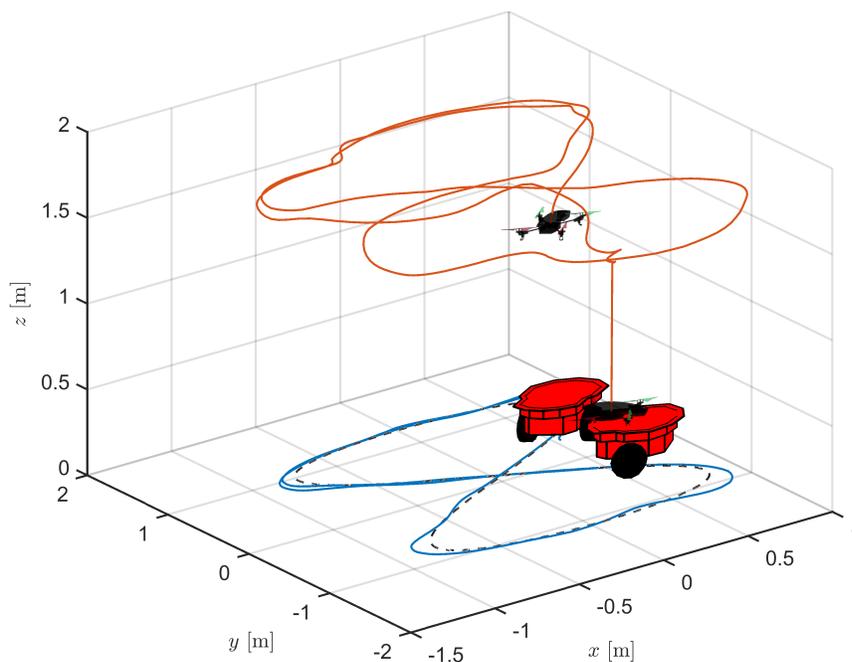


Figura 20 – Caminho percorrido pela formação quando  $V_{d_{max}} = 0,65 \text{ m/s}$ , com a estratégia de mudar  $V_d$  ao longo do caminho.

*Bebop 2* decola. Assim como na primeira parte, o desenho dos robôs representa o início e o final de cada parte da navegação. Além disso, uma linha contínua preta pode ser vista na segunda e na terceira partes, o que representa o último segundo da figura anterior, para melhor contextualizar a navegação. Esta parte mostra que o pouso do VANT foi bem sucedido, e o VTNT continuou seguindo o caminho proposto normalmente.

A terceira parte mostra a navegação da decolagem do VANT do VTNT em movimento até o seu pouso

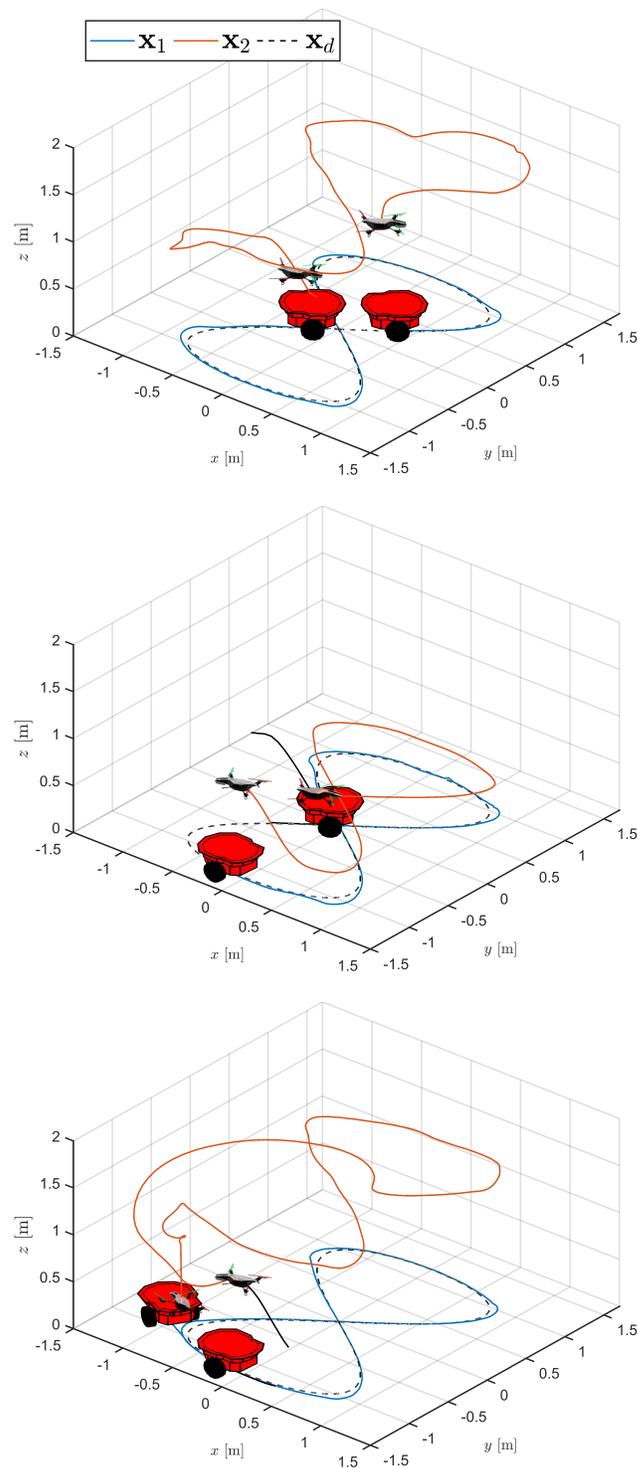


Figura 21 – Caminho percorrido pela formação durante o experimento, dividido em três partes.

final. Nos primeiros momentos fica claro que o VANT segue um arco muito maior que o VTNT. Isto ocorre porque enquanto o VANT ainda está no procedimento de decolagem, o VTNT não para de seguir o caminho na velocidade desejada, e tal fato faz com que os erros associados à forma da formação aumentem temporariamente. E, como tais variáveis de forma estão descritas em coordenadas esféricas, sua correção descreve este arco.

Após alguns segundos, os robôs estão novamente alinhados e, por fim, o VANT pousa sobre o VTNT. O vídeo disponível em <https://youtu.be/IPXWDFqnzSY?t=196> mostra o experimento completo, ou seja, as três partes acima descritas.

As Figuras 22 e 23 mostram os erros de formação e as velocidades do VTNT no experimento. Os retângulos azul claro na Figura 22 representam os momentos onde o VANT está pousado. Nestes momentos, não existem erros associados com a forma da formação, como é mostrado no lado direito da figura.

Finalmente, não é difícil notar que os erros associados a  $\alpha_f$  normalmente estão grandes. Isto ocorre pois quando  $\beta_f = \frac{\pi}{2}$ , como é o caso em  $\mathbf{q}_{d_n}$ , o VANT se encontra em um ponto de singularidade, onde pequenos movimentos do veículo causam uma grande variação em  $\alpha_f$ , mesmo que o robô ainda esteja essencialmente na mesma posição. Isto significa que quando o VANT se encontra neste ponto de singularidade os valores de  $\alpha_f$  não são relevantes para a forma, já que qualquer valor não irá alterá-la. Por outro lado, se forem considerados os momentos onde o robô aéreo está se preparando para pousar, quando a formação desejada é descrita por  $\mathbf{q}_{d_t}$ , tal veículo não se encontra mais na singularidade, e as Figuras 14, 18 e 22 mostram que após a linha que indica o pouso o controlador leva os valores de  $\tilde{\alpha}_f$  para valores próximos a zero. Soluções para tais singularidades fogem ao escopo desta Dissertação, mas se prevê tratá-las em trabalhos futuros.

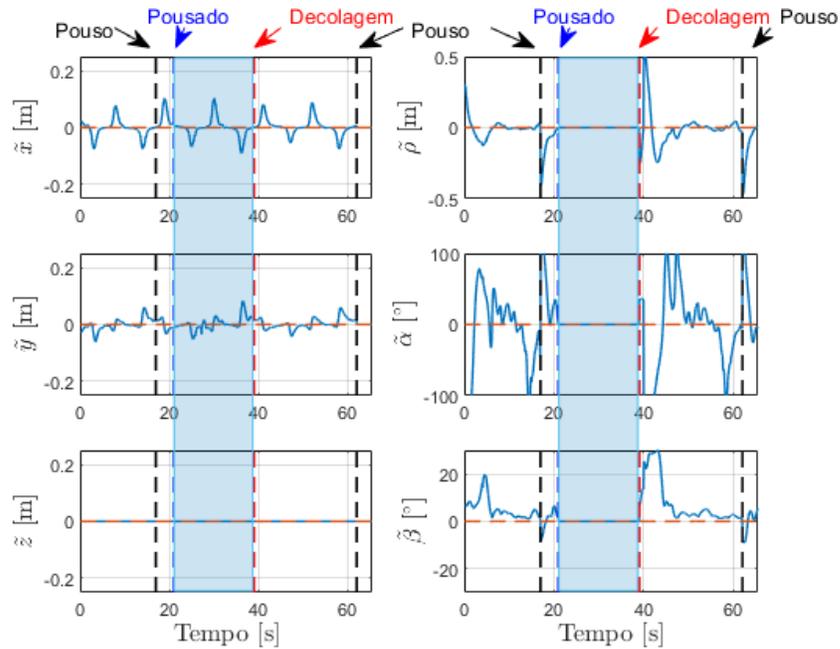


Figura 22 – Erros de formação quando o VANT pousa e decola de um VTNT em movimento.

Como observação final, é válido mencionar que todos os experimentos mostrados até agora usaram os mesmos ganhos, que estão listados em (4.4).

## 4.2 Seguimento de Caminho e Pousa do VANT sobre o VTNT Utilizando Espaço Nulo

Os experimentos a seguir ilustram o comportamento do seguimento de caminho pela formação e o pouso do VANT sobre o VTNT quando as tarefas parciais, a saber, mover a formação e manter a forma da estrutura virtual, são hierarquizadas de forma que a tarefa parcial de manter a forma da estrutura virtual tem maior prioridade. A decisão de dar prioridade à tarefa parcial de manter a forma da formação se deve à intenção de

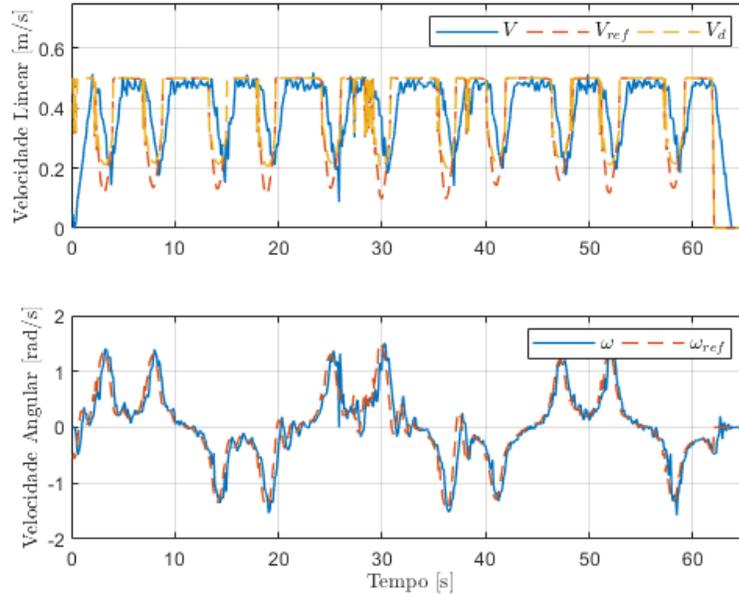


Figura 23 – Velocidades do VTNT para a tarefa de seguimento de caminho quando o VANT pouso e decola de um VTNT em movimento.

pousar o VANT sobre o VTNT, o que exige a correta posição relativa entre ambos. Como na seção anterior, será mostrado primeiro um pouso com o VTNT estático, e depois um pouso com o VTNT em movimento.

Além disso, para efeito de comparação, os experimentos mostrados nessa seção usam os mesmos ganhos e condições dos experimentos sem espaço nulo. A exceção é a duração do experimento de pouso estático, que mudou para 60 segundos, para mostrar de maneira mais completa o comportamento.

#### 4.2.1 Pouso Estático

O caminho a ser seguido é definido como

$$\begin{bmatrix} x(s) \\ y(s) \\ z(s) \end{bmatrix} = \begin{bmatrix} 0,8 \text{sen}(\frac{2\pi s}{100}) \text{ m} \\ 1,2 \text{sen}(\frac{\pi s}{100}) \text{ m} \\ 0 \text{ m} \end{bmatrix}, \quad (4.5)$$

enquanto a formação desejada, durante a navegação, é

$$\mathbf{q}_{d_n} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 1,2 \text{ m} \\ 0 \text{ rad} \\ \frac{\pi}{2} \text{ rad} \end{bmatrix}, \quad (4.6)$$

mudando para

$$\mathbf{q}_{d_i} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 0,75 \text{ m} \\ (\psi_1 + \pi) \text{ rad} \\ \frac{4\pi}{9} \text{ rad} \end{bmatrix}^T \quad (4.7)$$

quando vai ser iniciado o procedimento de pouso, o que é feito para assegurar que o VANT está centralizado sobre a plataforma em cima do VTNT. A formação navega seguindo o caminho determinado por 60 segundos, com  $V_{d_{max}} = 0,5 \text{ m/s}$ , e, após decorrido esse tempo, o VANT deve pousar sobre o VTNT.

Os resultados da navegação com o uso da técnica de Espaço Nulo para hierarquização de tarefas priorizando manter a forma da estrutura virtual são mostrados nas Figuras 24, 25 e 26. O vídeo deste experimento está disponível em <<https://youtu.be/qwzQiWxGOCM>>.

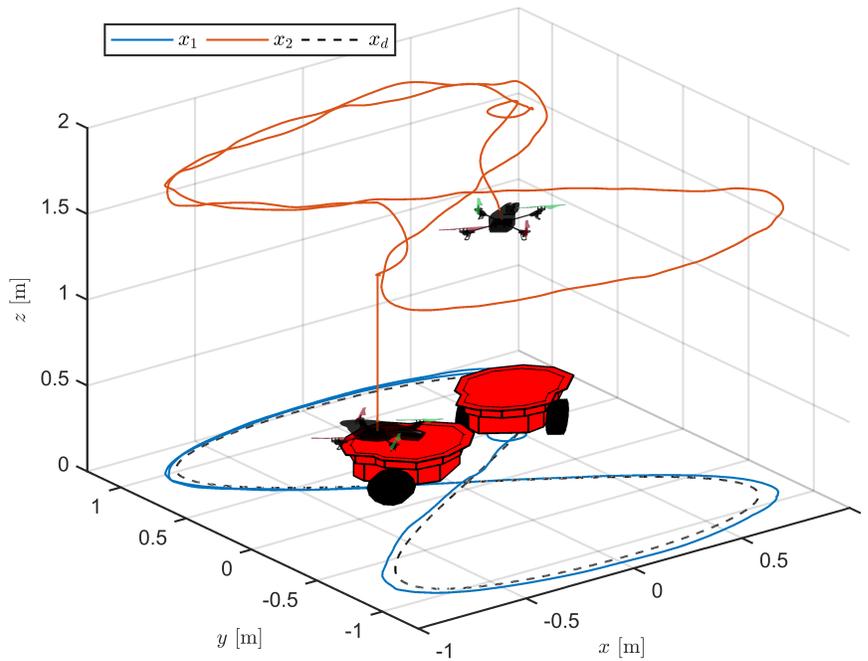


Figura 24 – Caminho percorrido pela formação quando há prioridade de forma.

Observando-se os erros de formação associados ao posicionamento do robô virtual percebe-se que eles são pequenos, ou seja, mesmo sendo priorizada a forma da estrutura virtual a tarefa de posicionamento também é cumprida. Enquanto isso, os erros de forma, especialmente  $\tilde{\rho}$  e  $\tilde{\beta}$ , mostram bom comportamento, ficando próximos de zero. Em relação ao erro de forma  $\tilde{\alpha}$ , fica evidente seus altos valores, comportamento este explicado na Seção 4.1.2.

Por último, a análise da Figura 26 revela que o robô terrestre não atinge  $V_d$ , ainda que siga seu padrão. Isto se deve ao fato de que as equações de Espaço Nulo obrigam os robôs, em primeiro lugar, a manter a forma da estrutura virtual, o que conflita com qualquer tentativa de atingir a velocidade desejada e mantê-la, pois a cada momento pequenas correções têm que ser feitas, para a manutenção da forma da estrutura virtual. Observe-se que, como dito no parágrafo anterior, os erros de formação relativos à posição são pequenos, ou seja, o robô não deixa em nenhum momento de seguir o caminho, ao priorizar a forma da estrutura virtual,

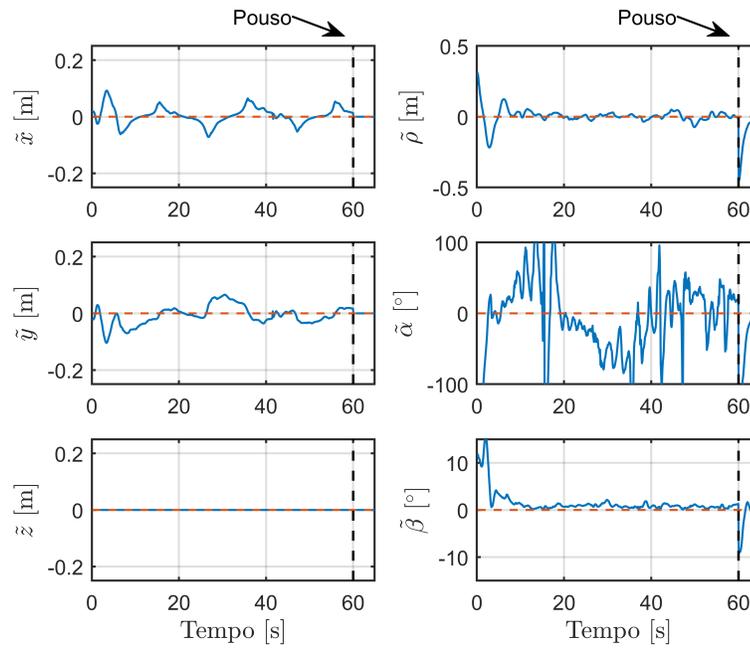


Figura 25 – Erros de formação quando há prioridade de forma.

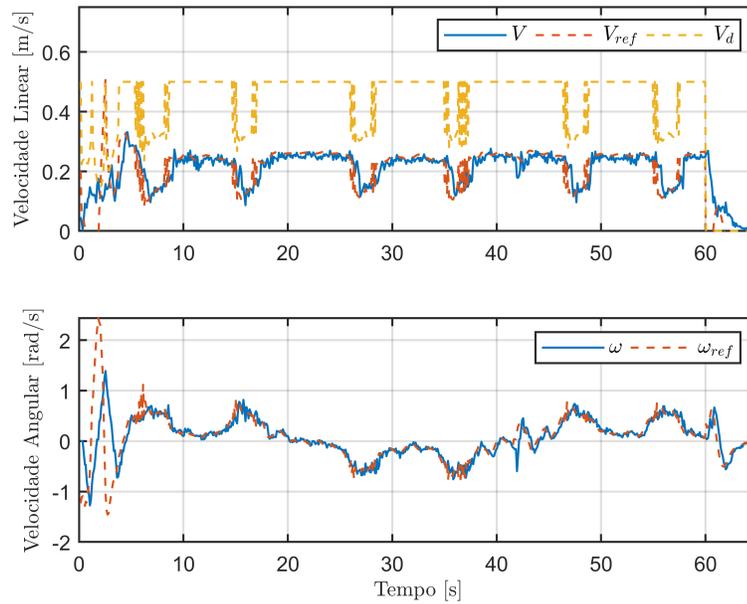


Figura 26 – Velocidades do VTNT quando há prioridade de forma.

mas existe um custo, que é a redução da velocidade ao longo do caminho. Qual a melhor estratégia a ser usada depende, então, da aplicação desejada para a formação.

### 4.2.2 Pouso em Movimento

O próximo experimento mostra uma tarefa de navegação da formação envolvendo a decolagem do VANT de cima de um VTNT estático, seguida por uma navegação da formação durante alguns segundos e o pouso enquanto o VTNT segue o caminho. Após algum tempo, o robô aéreo volta a decolar, desta vez enquanto o robô terrestre se move e, finalmente, após mais um tempo, o VTNT fica estático enquanto o VANT faz seu pouso final.

Para fazer isto, as formações desejadas  $\mathbf{q}_{d_n}$  e  $\mathbf{q}_{d_f}$  do experimento anterior foram utilizadas, com  $V_{d_{max}} = 0,50 \text{ m/s}$  para a parte de navegação e  $V_{d_{max}} = 0,0 \text{ m/s}$  para a parte estática da missão.

A Figura 27 mostra a navegação da formação durante o experimento, separada em três fases, para uma apresentação mais clara dos resultados. A primeira parte consiste do início da navegação até o momento do pouso em movimento. É possível observar nos robôs um ajuste inicial, onde o VTNT inicialmente segue fora do caminho. Isto acontece pois nestes momentos a formação está sendo corrigida, dada a maior prioridade em relação ao seguimento do caminho. Esses movimentos podem ser melhor observados no vídeo <<https://youtu.be/qwzQiWxGOcM?t=87>>.

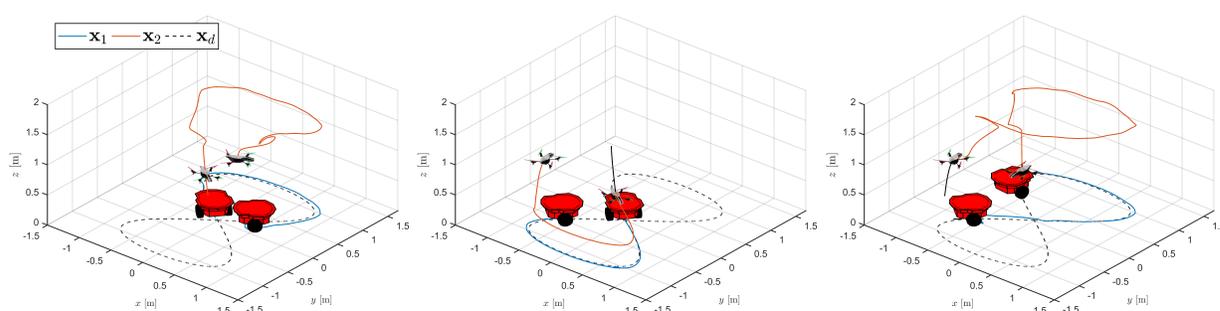


Figura 27 – Caminho percorrido pela formação durante o experimento, dividido em três partes.

A segunda parte mostra desde o pouso até o momento da segunda decolagem. Como na primeira parte, os desenhos dos robôs são mostrados nos momentos inicial e final de cada parte. Além disso, uma linha preta contínua ilustra o último segundo da fase anterior, para melhor contextualizar a navegação. Esta parte mostra que o pouso do VANT foi bem sucedido e o VTNT, por sua parte, continuou seguindo o caminho.

Por último, a terceira parte mostra a navegação a partir da segunda decolagem até o pouso final. Novamente, um rastro representado por uma linha preta mostra o último segundo da fase anterior. Observa-se que devido à decolagem do VANT aumentar consideravelmente os erros de forma, que são prioritários, os dois robôs se organizam para diminuir estes erros o mais rápido possível. O resultado é que o VTNT escapa momentaneamente do caminho seguido, mas fica evidente que após o estabelecimento da forma desejada o robô terrestre volta ao caminho sendo seguido, onde permanece até o pouso final.

Finalmente, as Figuras 28 e 29 mostram os erros de formação e as velocidades do VTNT, que coincide com o robô virtual da formação. Os retângulos azul claro na Figura 28 correspondem ao VANT aterrizado, situação em que não existem erros associados à forma da estrutura virtual, como mostrado na parte mais à direita da figura.

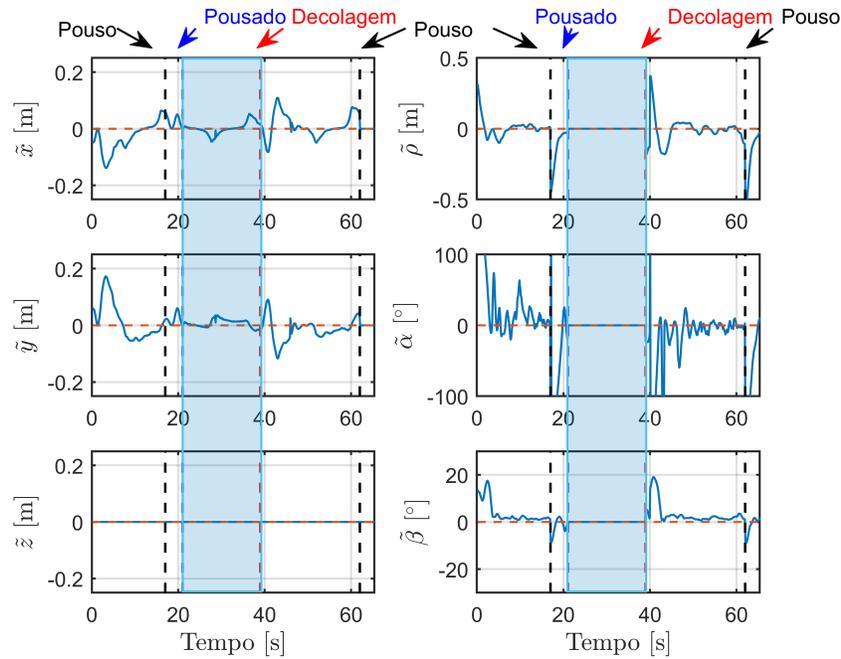


Figura 28 – Erros de formação para prioridade de forma.

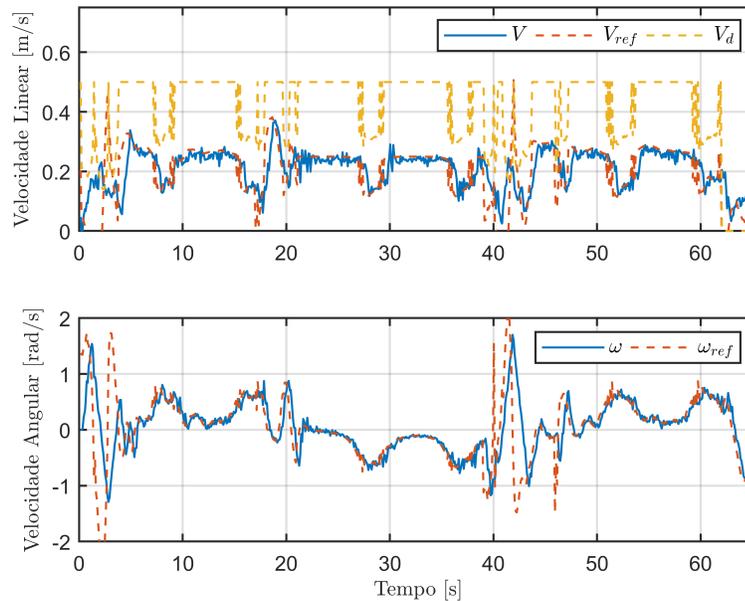


Figura 29 – Velocidades do VTNT com prioridade de forma.

### 4.3 Seguimento de Caminho e Pouso do VANT sobre o VTNT com Obstáculos

Para testar se o algoritmo de desvio de obstáculos proposto é uma solução válida foram feitas várias simulações. Na sequência, após a validação por meio de simulações, experimentos foram conduzidos para testar a eficiência e robustez do algoritmo em situações reais. Esta seção apresenta e discute os resultados obtidos em tais simulações e experimentos.

### 4.3.1 Simulações

O objetivo das simulações é validar a estratégia proposta para o desvio de obstáculos. Para alcançar isto, seis cenários foram construídos de maneira a mostrar os principais desafios que a formação pode encontrar, quando se considera a distribuição de obstáculos. A velocidade desejada da formação, em cada caso, foi  $V_{d_{max}} = 0,4 \text{ m/s}$ , enquanto os ganhos utilizados foram

$$\begin{aligned}
 \mathbf{L}_1 &= \text{diag} [1, 2 \quad 1, 2 \quad 3 \quad 1, 5 \quad 1, 5], \\
 \mathbf{L}_2 &= \text{diag} [1 \quad 1 \quad 1 \quad 1 \quad 1], \\
 L_3 &= 1, \\
 L_4 &= 1, \\
 \mathbf{L}_d &= \text{diag} [2 \quad 2 \quad 1, 8 \quad 5] \text{ e} \\
 L_c &= 1.
 \end{aligned} \tag{4.8}$$

Além disso, o deslocamento do ponto de controle para o centro do eixo das rodas no VTNT foi de  $a = 15 \text{ cm}$  e o tempo de amostragem foi de  $100 \text{ ms}$ , que é o tempo entre duas atualizações sucessivas das informações sensoriais do robô móvel a rodas *Pioneer 3-DX*, usado nos experimentos.

Quanto à formação desejada, esta foi caracterizada como

$$\mathbf{q}_d = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 1,5 \text{ m} \\ 0 \text{ rad} \\ \frac{\pi}{2} \text{ rad} \end{bmatrix}, \tag{4.9}$$

para cada caminho  $c(s)$  sendo seguido nos cenários.

Já em relação aos parâmetros do algoritmo de desvio, os raios de detecção e da zona de desvio foram definidos como  $r_{det} = 1,2 \text{ m}$  e  $r_{avd} = 0,8 \text{ m}$ . Um detalhe que vale mencionar é que a zona de desvio deve ser específica para cada obstáculo e pode ter tamanhos distintos para obstáculos distintos. Ainda que isto seja verdade, e o algoritmo consiga lidar naturalmente com tal mudança, desde que todos os  $r_{avd}$  sejam menores que  $r_{det}$ , para simplificar a discussão todos os obstáculos utilizaram o mesmo valor  $r_{avd}$  para caracterizar a zona de desvio.

Os cenários planejados são mostrados na Figura 30. A primeira coluna mostra os casos onde a formação encontra apenas um obstáculo de cada vez, ou seja, nos casos em que há múltiplos obstáculos eles estão longe o suficiente uns dos outros, de forma que cada um pode ser tratado como um único obstáculo. Estes casos serão discutidos primeiro, enquanto a discussão da segunda coluna (direita), onde a formação lida com múltiplos obstáculos ao mesmo tempo, será feita mais à frente.

A primeira figura mostra a formação enfrentando apenas um único obstáculo, representado pelo círculo vermelho, enquanto segue um caminho reto ao longo do eixo  $x$ . O VTNT começa de um ponto sobre o caminho desejado, enquanto o VANT começa de um ponto à sua direita. Rapidamente eles convergem para a formação desejada, desviam do obstáculo enquanto mantêm tal formação, e chegam ao final do caminho.

Vê-se que o caminho tomado foi o arco de circunferência mais curto em volta do obstáculo, já que o centro deste está abaixo do caminho ( $y_o$  é um número negativo), e a formação navega em pontos positivos de  $y$ . Além

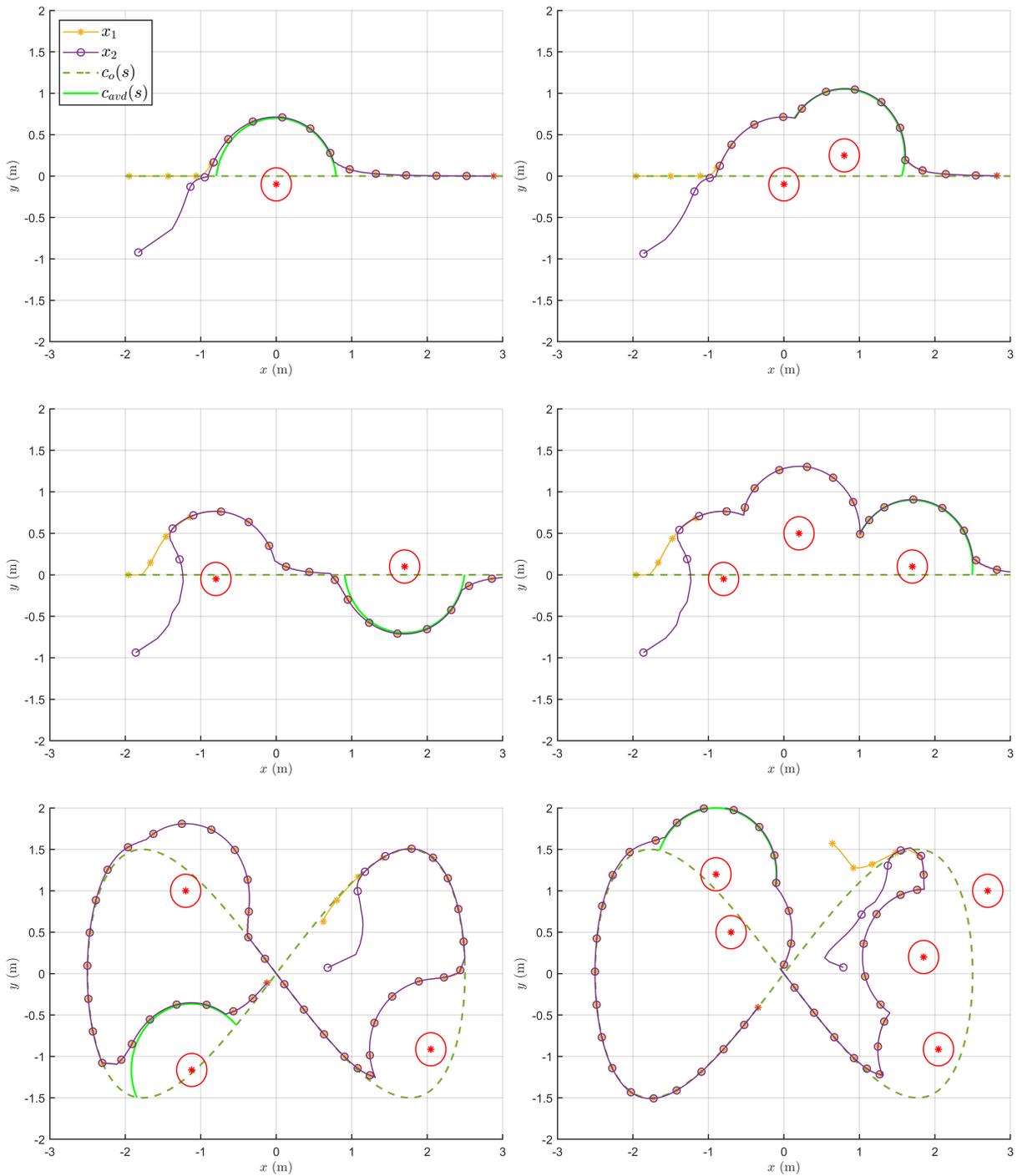


Figura 30 – Diferentes cenários imaginados para testar o algoritmo de desvio de obstáculos associado ao controlador de seguimento de caminho proposto.

disso, mesmo que possa parecer que o valor de  $r_{avd}$  é desnecessariamente grande para o tamanho do obstáculo, deve-se lembrar que apenas os pontos de controle dos veículos estão sendo mostrados. Na realidade o robô tem tamanho não nulo e isto deve ser levado em consideração, o que faz com que a zona de segurança deva ser grande o suficiente para permitir a passagem do corpo do robô (isto ficará mais claro nos experimentos).

O segundo cenário é parecido com o primeiro, exceto que agora existem dois obstáculos, um acima e outro abaixo do caminho. Novamente, o algoritmo seleciona o caminho mais curto em volta de cada obstáculo para continuar sua navegação. Em outras palavras, para o obstáculo que está abaixo do caminho, a formação navega acima dele, e para o obstáculo que está acima do caminho, a formação navega abaixo dele.

O terceiro cenário mostra um caminho mais complexo, onde o algoritmo deve determinar o caminho mais curto em volta dos obstáculos. O primeiro obstáculo é um caso interessante: em um primeiro momento parece ser melhor para a formação virar à esquerda (considerando sua orientação atual que, é virada para baixo) já que o obstáculo se encontra à direita do caminho. Entretanto, após o obstáculo, o caminho faz uma curva à direita e, por causa disso, o arco mais curto é na verdade para a direita do obstáculo, conforme mostrado na figura. Isto significa que o algoritmo escolheu de maneira efetiva o arco mais curto para evitar o obstáculo, mesmo em um caso um tanto mais complexo e genérico. Quanto aos outros obstáculos, eles foram evitados de maneira similar.

Até então, todos os casos discutidos mostram a formação evitando os obstáculos seguindo o caminho mais curto ao redor deles, o que está de acordo com o comportamento esperado do algoritmo para obstáculos únicos ou obstáculos que estão longe uns dos outros.

Os casos apresentados na segunda coluna da Figura 30 mostram múltiplos obstáculos próximos uns dos outros, o que requer o uso do algoritmo estendido. O cenário na figura superior direita mostra dois obstáculos que se encontram próximos um do outro. De maneira parecida com a versão de obstáculo singular, quando a formação encontra o primeiro obstáculo ela o evita tomando o caminho mais curto. Porém, durante a manobra de desvio a formação encontra um segundo obstáculo. Neste momento, se a formação escolhesse o caminho mais curto novamente é fácil perceber que ela colidiria com o primeiro obstáculo. Neste caso, é mostrado que o algoritmo escolhe o caminho mais longo para desviar do segundo obstáculo, já que este é o mais seguro. Após isso, a formação completa sua navegação.

Na segunda figura da coluna da direita da Figura 30 a formação tem que lidar com três obstáculos próximos. É possível observar que em cada caso o robô virtual seguiu o caminho mais seguro. Cada escolha depende das posições relativas de cada obstáculo, formação desejada e caminho planejado, como foi discutido, mas fica claro que o algoritmo tomou as decisões corretas.

Finalmente, o último cenário mostra dois aglomerados de obstáculos próximos. Para evitar o primeiro aglomerado, a formação inicialmente segue o caminho mais curto em volta do primeiro obstáculo, como de costume. Entretanto, quando chega no segundo obstáculo ela escolhe o caminho mais longo para retornar ao caminho original. Enquanto manobra para evitar o segundo obstáculo, a formação encontra o terceiro obstáculo do aglomerado, e prontamente muda sua direção para evitá-lo. No segundo aglomerado, um processo similar acontece e a formação navega ao seu redor de maneira bem sucedida.

Portanto, foi mostrado que o algoritmo se comporta da maneira projetada pela análise teórica. Ele faz com que a formação evite o primeiro obstáculo seguindo o arco mais curto ao redor dele e, em obstáculos subsequentes, ela seguirá o caminho mais seguro ao redor do obstáculo, quantas vezes forem necessárias até voltar ao caminho original.

Estes resultados de simulação mostram que a estratégia proposta realiza a tarefa de desvio de obstáculos em vários cenários diferentes, com desafios únicos, indicando que ela consegue lidar com distribuições genéricas de obstáculos no espaço de trabalho da formação.

### 4.3.2 Experimentos

Esta subsecção apresenta e analisa um experimento real onde a formação composta pelo VTNT e pelo VANT deve seguir um caminho correspondente a uma curva lemniscata de Bernoulli, ao longo da qual ela encontrará alguns obstáculos. O experimento foi feito usando a instalação descrita na Figura 11. Para obter as posições dos veículos e dos obstáculos o sistema de captura de movimento *OptiTrack* foi usado, configurado com oito câmeras espalhadas pela sala. Para comunicação com os robôs o *Robot Operating System - ROS* foi usado. Quanto ao código de controle de alto nível, este foi escrito em *Matlab*, enquanto a comunicação entre ROS e *Matlab* foi feita usando uma biblioteca proprietária do *Matlab*, que simula os nós e tópicos necessários para comunicação. Finalmente, a velocidade desejada máxima da formação,  $V_{d_{max}}$ , assim como os ganhos e parâmetros do algoritmo de desvio de obstáculos foram os mesmos utilizados nas simulações.

O caminho a ser seguido foi definido como

$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} -2, 0 \operatorname{sen}\left(\frac{\pi s}{180}\right) m \\ 1, 2 \operatorname{sen}\left(\frac{2\pi s}{180}\right) m \end{bmatrix}. \quad (4.10)$$

Já a formação desejada utilizada é caracterizada por

$$\mathbf{q}_{d_n} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 1,5 m \\ 0 \text{ rad} \\ \frac{\pi}{2} \text{ rad} \end{bmatrix}, \quad (4.11)$$

para a parte do experimento onde os dois veículos navegam como uma formação, e

$$\mathbf{q}_{d_l} = \begin{bmatrix} c_{n_x} \\ c_{n_y} \\ c_{n_z} \\ 0,8 m \\ (\psi_1 + \pi) \text{ rad} \\ \frac{4\pi}{9} \text{ rad} \end{bmatrix}, \quad (4.12)$$

para a parte do experimento onde o VANT se prepara para pousar sobre o VTNT, sendo  $\psi_1$  a orientação do VTNT.

A tarefa a ser cumprida pela formação é seguir o caminho descrito na Eq. (4.10), reagindo ao surgimento de qualquer obstáculo em seu caminho. Durante a primeira volta não há nenhum obstáculo a ser evitado, de maneira a estabelecer o padrão de navegação normal. Na segunda volta, dois obstáculos são colocados no caminho, longe um do outro. O objetivo é mostrar o algoritmo funcionando no desvio de um único obstáculo de cada vez. Finalmente, na terceira e última volta da formação ao longo do caminho escolhido, três obstáculos são colocados próximos uns aos outros, em posições escolhidas aleatoriamente ao longo do caminho. Todos eles foram evitados pela formação, como esperado.

O experimento completo do controlador de seguimento de caminhos com desvio de obstáculos pode ser acompanhado assistindo-se ao vídeo disponível no *link* <[https://youtu.be/Zkaxw\\_hxkwg](https://youtu.be/Zkaxw_hxkwg)>.

Para ajudar na análise do experimento a Figura 31 sintetiza a navegação. Em tal figura, de maneira a facilitar o entendimento da operação do sistema proposto, a navegação foi dividida em três partes, cada parte consistindo de uma volta no caminho. Na primeira volta os dois robôs rapidamente convergem para

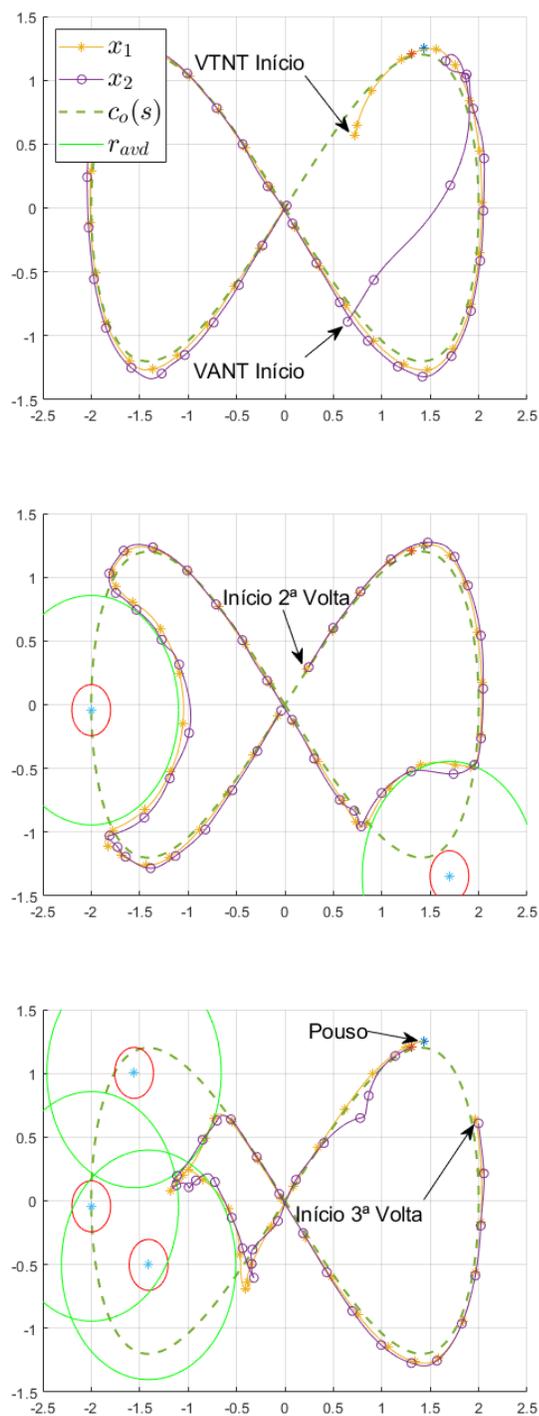


Figura 31 – Caminho realizado pela formação durante a navegação, dividido em três partes.

suas posições correspondentes, para a montagem da formação. Depois disso, a formação segue o caminho especificado sem problemas, como mostra a primeira sub figura.

Na segunda volta, a formação encontra um obstáculo na segunda curva à direita, como mostrado na segunda sub figura. Como pode ser visto, a formação desvia de tal obstáculo seguindo um caminho temporário

seguro. Depois, logo após virar à esquerda, um novo obstáculo faz com que a formação siga um novo caminho temporário seguro, gerado através do algoritmo de desvio. Na última parte da Figura 31, a volta deveria começar no mesmo ponto onde a segunda volta termina. Porém, como esta era uma zona livre de obstáculos e o pouso aconteceu nesta região, o início da terceira volta foi omitido, e um ponto de início fora da região de pouso é mostrado no gráfico. A formação começa seguindo o caminho normalmente na terceira volta, até encontrar os três obstáculos, logo antes de virar à esquerda pela primeira vez. Ao encontrá-los, a formação usa os caminhos temporários gerados pelo algoritmo e navega de maneira segura para um ponto além dos obstáculos, de onde continua o seguimento do caminho original.

Finalmente, ao final da navegação, o VANT pouso no VTNT ainda em movimento, e então a navegação como um todo termina.

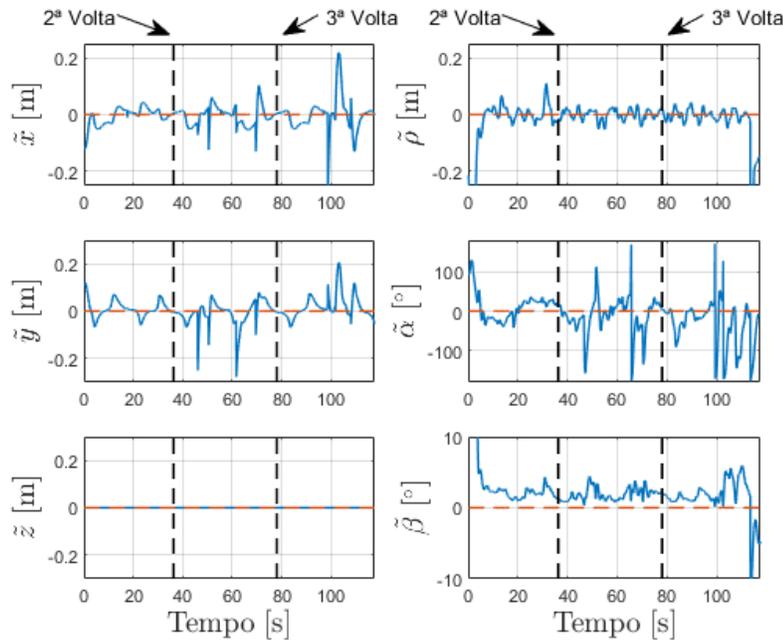


Figura 32 – Erros de formação.

Os erros de formação durante a navegação são mostrados na Figura 32, onde os pontos iniciais da segunda e terceira volta são mostrados. Em relação a  $\tilde{x}$  e  $\tilde{y}$ , é aparente que os picos de erro são menores na primeira volta, um pouco maiores na segunda e maiores na terceira. Isto ocorre porque a mudança súbita do caminho inicialmente desejado para um caminho temporário seguro para desvio dos obstáculos aumenta o erro momentaneamente. Entretanto, é possível observar que o controlador é capaz de reduzir tais erros, e que a magnitude máxima destes erros é da ordem de 20 centímetros. O erro no eixo  $z$ ,  $\tilde{z}$ , é sempre zero, pois se considera que o robô virtual, que corresponde à referência da formação, e que coincide com o VTNT, sempre navega no plano  $xy$ , onde  $z = 0$ .

A segunda coluna da Figura 32 retrata os erros de forma da formação. O erro associado com o tamanho da linha 3D,  $\tilde{\rho}$ , é rapidamente reduzido à um valor próximo a zero, o que corresponde à composição da formação desejada, e então é mantido sob controle. O erro em  $\tilde{\alpha}$  exibe comportamento parecido. Já o erro em  $\tilde{\beta}$  tem comportamento parecido com aquele da Seção 4.1, onde se explicam as razões para esse comportamento.

A Figura 33 compara as velocidades desejadas do VTNT, que representa o robô virtual que serve de referência para a formação, com os valores medidos. O primeiro gráfico mostra a velocidade linear real,

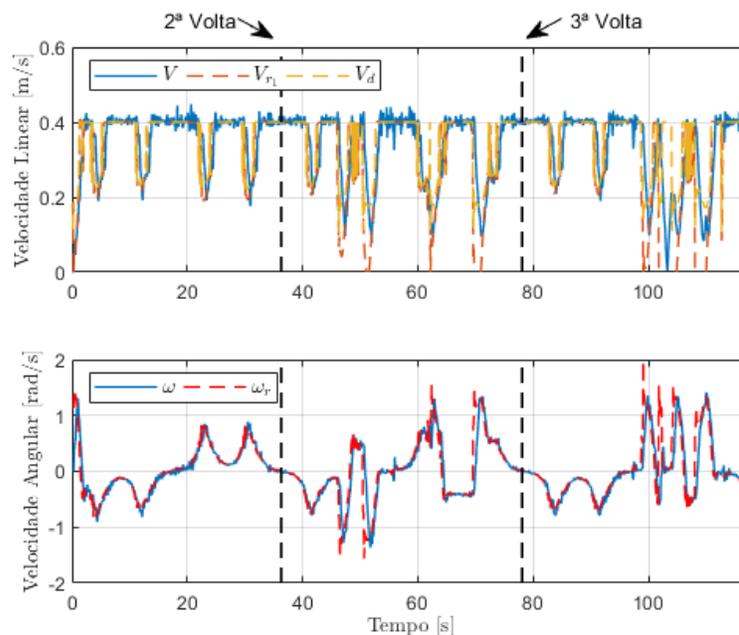


Figura 33 – Velocidades linear e angular.

em uma linha azul contínua, e a saída do controlador, em uma linha vermelha tracejada, enquanto a linha amarela representa a velocidade desejada  $V_d$  ao longo do caminho. Os resultados mostram que o seguimento da velocidade desejada é quase perfeito, já que as três curvas são bastante similares. Ainda assim, é possível perceber pequenos atrasos sempre que há uma queda repentina na velocidade de referência, que são causadas pela estratégia de redução de velocidade em curvas acentuadas, discutida na Subseção 2.4.4. Tais atrasos ocorrem devido à falta de um compensador dinâmico para o VTNT, que lidaria com o erro de seguimento de velocidade. Porém, mesmo sem o compensador, os atrasos são pequenos o suficiente para serem desprezados na presente aplicação.

Finalmente, devido às restrições não holonômicas do VTNT, sua velocidade angular não pode ser escolhida livremente, como o é a velocidade linear. A velocidade angular é uma consequência da configuração do caminho, já que o robô se move sempre para frente (ou para trás), sobre o caminho. Por causa disso, não há valor de referência para a velocidade angular. Entretanto, o seu controle correto ainda é vital, pois senão o robô não conseguirá se mover na direção desejada. Felizmente, é possível observar que o robô seguiu de maneira bem sucedida as referências de saída do controlador para velocidade angular. Portanto, como o robô seguiu com sucesso ambas as velocidades linear e angular, tanto a magnitude quanto a direção da velocidade desejada ao longo do caminho estão corretas.



## 5 Conclusão

Este trabalho propõe um controlador projetado para guiar uma formação heterogênea de robôs (um robô terrestre e um aéreo) em uma tarefa de seguimento de caminho, além de explorar conceitos de espaço nulo e desvio de obstáculos. Para cumprir tal objetivo, a formação foi caracterizada e o controlador foi desenvolvido, e, posteriormente, adaptado para incluir tanto a técnica de espaço nulo quanto o algoritmo de desvio de obstáculos. Ainda considerando o controlador de seguimento de caminho, uma estratégia de redução de velocidade em curvas acentuadas foi desenvolvida e aplicada, o que permite ao usuário configurar uma velocidade máxima para o caminho, ao invés de uma que seja viável em todos os pontos. Nesse sentido, é importante frisar que o controlador proposto não apresenta singularidades em termos de velocidade ao longo do caminho, o que permite o ajuste da mesma e, inclusive, a parada da formação quando necessário.

Para a inclusão da técnica de espaço nulo, a tarefa de navegação foi considerada em duas partes, a saber, manter a forma da estrutura virtual que descreve a formação, a linha reta que une os dois veículos, e mover a formação ao longo do caminho. Prioridades diferentes foram dadas a essas duas tarefas parciais, sendo que a tarefa de manter a forma da estrutura virtual teve a maior prioridade. A razão para isso é que se considera que o VANT vai pousar sobre o VTNT, o que exige que ele esteja muito bem posicionado em relação ao VTNT, de maneira a assegurar um pouso sem problemas. Isso quer dizer que é necessário manter a forma da estrutura virtual a qualquer custo, mesmo que isso signifique, momentaneamente, deixar de seguir o caminho.

Em outro momento, um algoritmo para desvio de obstáculos foi formalizado, primeiro considerando apenas obstáculos únicos. Em seguida tal algoritmo foi estendido para lidar com múltiplos obstáculos de vez. Isto é feito estabelecendo zonas de desvio ao redor dos obstáculos, modeladas como círculos, e então navegando em caminhos temporários seguros estabelecidos pelo algoritmo, até chegar em outro ponto do caminho original localizado após os obstáculos. No caso de obstáculos únicos, o caminho seguro é sempre o menor arco do círculo delimitando a zona de desvio, já que ambos os arcos são igualmente seguros. Por outro lado, para múltiplos obstáculos, a navegação será feita pelo arco de circunferência mais seguro, seja ele o mais curto ou o mais longo, e isto será definido em tempo real através de uma análise caso a caso.

Com os resultados apresentados, que envolvem tanto simulações quanto experimentos, é possível afirmar que o controlador é efetivo em guiar a formação ao longo do caminho e que, utilizando o algoritmo de desvio proposto, a formação é capaz de navegar de maneira segura em um ambiente com obstáculos previamente desconhecidos.

Como trabalho futuro, está sob estudo modificar o algoritmo de múltiplos obstáculos para modelar os obstáculos como uma única elipse, ao invés de modelar zonas de desvio separadas como círculos. A ideia é que isto pode melhorar a navegação, dentro do espaço disponível. Além disso, o método usado, a saber, levar os pontos de interesse para um novo sistema de referência, centralizado no obstáculo, faz com que tal modificação seja mais natural, pois enquanto o círculo é imune a rotações em relação ao seu centro, uma elipse não é, o que faz com que sua definição, e a definição de seus arcos curto e longo sejam definitivamente mais simples quando tal figura estiver localizada na origem, sem rotação.

Como um segundo desafio, está sendo estudado como lidar com obstáculos móveis. Neste sentido, a habilidade do controlador de mudar livremente a velocidade da formação é uma grande vantagem, já que em muitos casos a abordagem mais natural pode ser parar os robôs e esperar sobre o caminho até que os obstáculos móveis não representem mais um risco para a navegação. Ou até mesmo mudar sua velocidade de tal forma que a formação e o obstáculo móvel não tenham interseção no caminho simultaneamente.



## Referências

- ANTONELLI, G.; ARRICHIELLO, F.; CHIAVERINI, S. The null-space-based behavioral control for autonomous robotic systems. *Intelligent Service Robotics*, Springer, v. 1, n. 1, p. 27–39, 2008. Citado 2 vezes nas páginas 24 e 40.
- BACHETI, V. P.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Seguimento de caminho usando espaço nulo para uma formação vant-vtnt. *Anais da Sociedade Brasileira de Automática*, v. 2, n. 1, 2020. Citado na página 24.
- BACHETI, V. P.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Path-following with a ugv-uav formation considering that the uav lands on the ugv. In: *Proceedings of the 2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. Athens, Greece: [s.n.], 2020. p. 488–497. Citado na página 39.
- BAI, H.; WEN, J. T. Cooperative load transport: A formation-control perspective. *IEEE Transactions on Robotics*, v. 26, n. 4, p. 742–750, August 2010. ISSN 1552-3098. Citado na página 24.
- 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, Jun 2015. ISSN 2195-3899. Citado na página 33.
- CERQUEIRA, T. A.; SANTOS, T. L.; CONCEICAO, A. G. A new approach based in potential fields with obstacles avoidance for mobile robots. In: *IEEE. 2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*. [S.l.], 2016. p. 229–233. Citado na página 25.
- DAI, S.; LEE, T.; BERNSTEIN, D. S. Adaptive control of a quadrotor uav transporting a cable-suspended load with unknown mass. In: *53rd IEEE Conference on Decision and Control*. [S.l.: s.n.], 2014. p. 6149–6154. ISSN 0191-2216. Citado na página 24.
- DAS, P. kumar; KONAR, A.; LAISHRAM, R. Path planning of mobile robot in unknown environment. In: *Special Issue of IJCCT for International Conference [ACCTA-10]*. [S.l.: s.n.], 2010. v. 1, n. 2, p. 3–5. Citado na página 25.
- DONG, X. et al. Theory and experiment on formation-containment control of multiple multirotor unmanned aerial vehicle systems. *IEEE Transactions on Automation Science and Engineering*, v. 16, n. 1, p. 229–240, January 2019. ISSN 1558-3783. Citado 2 vezes nas páginas 24 e 26.
- ERNANDES-NETO, V.; SARCINELLI-FILHO, M.; BRANDÃO, A. S. Trajectory-tracking of a heterogeneous formation using null space-based control. In: *IEEE. Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2019. p. 187–195. Citado na página 24.
- ERNANDES-NETO, V.; SARCINELLI-FILHO, M.; BRANDÃO, A. S. Trajectory-tracking of a heterogeneous formation using null space-based control. In: *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. Atlanta, GA, USA: [s.n.], 2019. p. 187–195. ISSN 2373-6720. Citado na página 26.
- JIANG, B. W. et al. Thrust vectoring control for infrastructure inspection multirotor vehicle. In: *2019 IEEE 6th International Conference on Industrial Engineering and Applications (ICIEA)*. [S.l.: s.n.], 2019. p. 209–213. ISSN null. Citado 2 vezes nas páginas 24 e 25.
- JORDAN, S. et al. State-of-the-art technologies for uav inspections. *IET Radar, Sonar Navigation*, v. 12, n. 2, p. 151–164, 2018. ISSN 1751-8792. Citado 2 vezes nas páginas 24 e 25.
- KHALIL, H. *Nonlinear Systems*. [S.l.]: Prentice Hall, 2002. (Pearson Education). ISBN 9780130673893. Citado na página 39.
- KIM, J. H.; KWON, J.-W.; SEO, J. Multi-uav-based stereo vision system without gps for ground obstacle mapping to assist path planning of ugv. *Electronics Letters, IET*, v. 50, n. 20, p. 1431–1432, 2014. Citado na página 24.

- 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, 2009. Citado na página 24.
- 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 1941-014X. Citado 2 vezes nas páginas 33 e 34.
- KOVÁCS, B. et al. A novel potential field method for path planning of mobile robots by adapting animal motion attributes. *Robotics and Autonomous Systems*, Elsevier, v. 82, p. 24–34, 2016. Citado na página 25.
- KWAŚNIEWSKI, K. K.; GOSIEWSKI, Z. Genetic algorithm for mobile robot route planning with obstacle avoidance. *acta mechanica et automatica*, Sciendo, v. 12, n. 2, p. 151–159, 2018. Citado 2 vezes nas páginas 25 e 26.
- LEWIS, M. A.; TAN, K.-H. High precision formation control of mobile robots using virtual structures. *Autonomous Robots*, v. 4, n. 4, p. 387–403, October 1997. ISSN 1573-7527. Citado 2 vezes nas páginas 24 e 32.
- LIU, Y.-C.; LIU, S.-Y.; WANG, N. Fully-tuned fuzzy neural network based robust adaptive tracking control of unmanned underwater vehicle with thruster dynamics. *Neurocomputing*, Elsevier, v. 196, p. 1–13, 2016. Citado na página 25.
- LOW, C. B. A flexible virtual structure formation keeping control design for nonholonomic mobile robots with low-level control systems, with experiments. In: IEEE. *Intelligent Control (ISIC), 2014 IEEE International Symposium on*. [S.l.], 2014. p. 1576–1582. Citado na página 32.
- LYU, Y. et al. Simultaneously multi-uav mapping and control with visual servoing. In: IEEE. *Proceedings of the 2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2015. p. 125–131. Citado na página 24.
- 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*. Daejeon, South Korea, 2016. p. 1623–1630. Citado na página 24.
- MATHIAS, H. D.; RAGUSA, V. R. An empirical study of crossover and mass extinction in a genetic algorithm for pathfinding in a continuous environment. In: IEEE. *2016 IEEE Congress on Evolutionary Computation (CEC)*. [S.l.], 2016. p. 4111–4118. Citado na página 25.
- MEISSEN, C. et al. Passivity-based formation control for uavs with a suspended load. *IFAC-PapersOnLine*, Elsevier, v. 50, n. 1, p. 13150–13155, 2017. Citado 2 vezes nas páginas 24 e 26.
- MOREIRA, M. S. M.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Null space based formation control for a uav landing on a ugv. In: IEEE. *Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2019. p. 1389–1397. Citado na página 24.
- MOREIRA, M. S. M.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Null space based formation control for a uav landing on a ugv. In: *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. Atlanta, GA, USA: [s.n.], 2019. p. 1389–1397. ISSN 2373-6720. Citado na página 26.
- MURUGAN, D.; GARG, A.; SINGH, D. Development of an adaptive approach for precision agriculture monitoring with drone and satellite data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, v. 10, n. 12, p. 5322–5328, December 2017. ISSN 2151-1535. Citado 2 vezes nas páginas 24 e 25.
- PALUNKO, I.; CRUZ, P.; FIERRO, R. Agile load transportation: Safe and efficient load manipulation with aerial robots. *IEEE Robotics Automation Magazine*, v. 19, n. 3, p. 69–79, September 2012. ISSN 1558-223X. Citado na página 24.
- PARKER, L. E. Multiple Robot Systems. In: BRUNO SICILIANO AND OUSSAMA KHATIB. *Springer Handbook of Robotics*. 2nd. ed. [S.l.]: Springer International Publishing, 2016. (Springer Handbooks), p. 1335–1384. ISBN 978-3-319-32552-1. Citado na página 24.

- PINTO, A. O. et al. High-level modeling and control of the *Bebop 2* micro aerial vehicle. In: *To appear in the Proceedings of the 2020 International Conference on Unmanned Aircraft Systems*. Athens, Greece: [s.n.], 2020. Citado 4 vezes nas páginas 31, 32, 34 e 39.
- PIZETTA, I. H. B.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Load transportation by quadrotors in crowded workspaces. *IEEE Access*, v. 8, p. 223941–223951, 2020. Citado 2 vezes nas páginas 24 e 26.
- PIZETTA, I. H. B.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. . Cooperative load transportation using three quadrotors. In: *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. Atlanta, GA, USA: [s.n.], 2019. p. 644–650. ISSN 2373-6720. Citado 2 vezes nas páginas 24 e 26.
- POTHAL, J. K.; PARHI, D. R. Navigation of multiple mobile robots in a highly clutter terrains using adaptive neuro-fuzzy inference system. *Robotics and Autonomous Systems*, Elsevier, v. 72, p. 48–58, 2015. Citado na página 25.
- QIN, H. et al. Autonomous exploration and mapping system using heterogeneous uavs and ugvs in gps-denied environments. *IEEE Transactions on Vehicular Technology*, v. 68, n. 2, p. 1339–1350, 2019. Citado 2 vezes nas páginas 24 e 26.
- RABELO, M. F. S.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Landing a uav on static or moving platforms using a formation controller. *IEEE Systems Journal*, p. 1–9, 2020. Citado 5 vezes nas páginas 24, 32, 33, 36 e 37.
- RANIERI, L. et al. A review of last mile logistics innovations in an externalities cost reduction vision. *Sustainability*, Multidisciplinary Digital Publishing Institute, v. 10, n. 3, p. 782, 2018. Citado na página 27.
- RESENDE, C. Z. et al. A new positioning and path following controller for unicycle mobile robots. In: *IEEE. 2013 16th International Conference on Advanced Robotics (ICAR)*. [S.l.], 2013. p. 1–6. Citado na página 34.
- RESENDE, C. Z.; CARELLI, R.; SARCINELLI-FILHO, M. Coordinated path-following for multi-robot systems using the cluster space framework approach. In: *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*. [S.l.: s.n.], 2014. p. 332–337. ISSN 2378-363X. Citado na página 33.
- RUBIO, F.; VALERO, F.; LLOPIS-ALBERT, C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. *International Journal of Advanced Robotic Systems*, SAGE Publications Sage UK: London, England, v. 16, n. 2, p. 1729881419839596, 2019. Citado na página 23.
- SANTANA, L. V. et al. A trajectory tracking and 3d positioning controller for the AR.Drone quadrotor. In: *IEEE. 2014 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2014. p. 756–767. Citado na página 31.
- SANTANA, L. V.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Navigation and cooperative control using the ar.drone quadrotor. *Journal of Intelligent Robotic Systems*, v. 84, n. 1, p. 327–350, 2016. Citado na página 31.
- SANTOS, M. C. P. et al. A novel null-space-based uav trajectory tracking controller with collision avoidance. *IEEE/ASME Transactions on Mechatronics*, v. 22, n. 6, p. 2543–2553, Dec 2017. ISSN 1941-014X. Citado na página 25.
- SANTOS, M. C. P. et al. A novel null-space-based uav trajectory tracking controller with collision avoidance. *IEEE/ASME Transactions on Mechatronics*, IEEE, v. 22, n. 6, p. 2543–2553, 2017. Citado na página 26.
- SHAKHATREH, H. et al. Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges. *IEEE Access*, v. 7, p. 48572–48634, 2019. ISSN 2169-3536. Citado na página 24.
- SHARMA, V.; KUMAR, R. A cooperative network framework for multi-uav guided ground ad hoc networks. *Journal of Intelligent & Robotic Systems*, Springer, v. 77, n. 3-4, p. 629–652, 2015. Citado na página 24.
- SHIRANI, B.; NAJAFI, M.; IZADI, I. Cooperative load transportation using multiple uavs. *Aerospace Science and Technology*, Elsevier, v. 84, p. 158–169, 2019. Citado na página 24.

- SIEGWART, R. Nourbakhsh, introduction to autonomous mobile robots. *hc: alk. paper*, v. 1, p. 13–36, 2004. Citado na página 23.
- SOLOVEY, E. T.; RYAN, K. J.; CUMMINGS, M. Coda: Mobile interface for enabling safer navigation of unmanned aerial vehicles in real-world settings. *International Journal of Human-Computer Studies*, Elsevier, v. 145, p. 102508, 2020. Citado 2 vezes nas páginas 24 e 25.
- 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 24.
- TZAFESTAS, S. G. Mobile robot path, motion, and task planning. *Introduction to Mobile Robot Control*, Elsevier, p. 429–478, 2014. Citado na página 25.
- VILLA, D. K. D.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Rod-shaped payload transportation using multiple quadrotors. In: IEEE. *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.], 2019. p. 1036–1040. Citado 2 vezes nas páginas 24 e 26.
- VILLA, D. K. D.; BRANDÃO, A. S.; SARCINELLI-FILHO, M. Outdoor navigation using two quadrotors and adaptive sliding mode control. In: *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. [S.l.: s.n.], 2020. p. 716–721. Citado na página 30.
- YAN, Z. et al. A real-time reaction obstacle avoidance algorithm for autonomous underwater vehicles in unknown environments. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 18, n. 2, p. 438, 2018. Citado na página 25.
- ZHANG, C.; KOVACS, J. M. The application of small unmanned aerial systems for precision agriculture: a review. *Precision agriculture*, Springer, v. 13, n. 6, p. 693–712, 2012. Citado na página 24.