UNIVERSIDADE FEDERAL DE SERGIPE

PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

PLANEJAMENTO DE MOVIMENTO AGRESSIVO PARA UM SISTEMA DE QUADRICÓPTERO COM CARGA SUSPENSA BASEADO EM RRT

JEFFERSON DE LIMA SILVEIRA JÚNIOR

São Cristóvão-SE, Brasil

Março de 2020

PLANEJAMENTO DE MOVIMENTO AGRESSIVO PARA UM SISTEMA DE QUADRICÓPTERO COM CARGA SUSPENSA BASEADO EM RRT

JEFFERSON DE LIMA SILVEIRA JÚNIOR

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica – PROEE, da Universidade Federal de Sergipe, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica

Orientador: Sidney Nascimento Givigi Júnior
Co-orientador: Eduardo Oliveira Freire

São Cristóvão-SE, Brasil

Março de 2020

UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
COORDENAÇÃO DE PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA-PROEE

TERMO DE APROVAÇÃO

" Planejamento de movimento agressivo para um sistema de quadricóptero com carga suspensa baseado em RRT "

Discente:

Jefferson de Lima Silveira Júnior

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Sergipe, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Aprovada pela banca examinadora composta por:

Prof. Dr. Jugurta Rosa Montalvão Filho (PROEE/UFS)
Presidente

Prof. Dr. Lucas Molina (DEL/UFS)
Examinador Externo

Prof. Dr. José Gilmar Nunes de Carvalho Filho (DEL/UFS)
Examinador Externo

Prof. Dr. Jeremias Barbosa Machado (UNIFEI)
Examinador externo

Jefferson de Lima Silveira Júnior
Candidato

Cidade Universitária "*Prof. José Aloísio de Campos*", 03 de março de 2020.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE

Silveira Júnior, Jefferson de Lima
S587p Planejamento de movimento agressivo para um sistema de quadricóptero com carga suspensa baseado RRT / Jefferson de Lima Silveira Júnior ; orientador Sidney Nascimento Givigi Júnior. - São Cristóvão, 2020.
91 f. : il.

Dissertação (mestrado em Engenharia Elétrica) – Universidade Federal de Sergipe, 2020.

1. Engenharia Elétrica. 2. Drone. 3. Robótica. 4. Planejamento. I. Givigi Júnior, Sidney Nascimento, orient. II. Título.

CDU 621.3

Agradecimentos

Gostaria de agradecer primeiramente ao meu pai Jefferson e à minha mãe Inez por terem sempre se dedicado ao máximo na educação de seus filhos. Sem dúvidas, eu me encontro nesta posição devido ao alicerce construído ao longo dos anos em uma família que sempre me apoiou em todas minhas decisões. Agradeço aos meus irmãos Keicyane, Jonatas e Karinne.

À família Hora que sempre se mantém unida e presente. Em todas as minhas conquistas, minha família sempre esteve presente torcendo e me apoiando no que fosse necessário. Pelos finais de semana de encontro, que são sempre alegres e cheios de bons conselhos.

Ao meu orientador Sidney Givigi, que vem acreditando em mim e fornecendo discussões extremamente edificantes e por ter disponibilizado bastante de seu tempo durante minhas dificuldades durante o mestrado. Ao meu co-orientador, Eduardo Freire, que vem trabalhando comigo desde que eu me conheço como pesquisador. Muito obrigado por toda a dedicação. Não só no mestrado, mas em toda minha formação na área de robótica.

Ao professor Lucas Molina, que sempre esteve presente com suas observações críticas mas sempre esclarecedoras e que sempre ajudaram a valorizar mais meus trabalhos. Gostaria de agradecer também pelas discussões que sempre surgem nos corredores sobre a vida e sobre rótulos. Ao professor Elyson Carvalho pelas orientações em todas as áreas possíveis da vida. Por ser muito mais que um professor e orientador.Ao professor Gilmar com as conversas sempre interessantes. Pela sua capacidade em sugerir pontos que sempre ajudam a melhorar as contribuições e a valorizar nossos trabalhos. Ao professor Jugurta, que sempre surge com as discussões mais profundas sobre o conhecimento e principalmente pela maneira única que ele tem de enxergar o mundo.

A todo o pessoal do GPRUFS que sempre alegram o laboratório e estão sempre presentes nas discussões das mais aleatórias às mais essenciais.

Gostaria também de agradecer à Graziella Bedenik por todo amor e carinho. Por sempre se manter presente. Por entender os momentos de dificuldades. Por trabalhar ao meu lado nos dias mais ocupados. Por alegrar os meus dias. Por desejar o meu crescimento e principalmente por querer participar da minha vida com tanto fervor.

Também gostaria de agradecer a Deus pela saúde me dada.

Por fim, agradeço ao apoio financeiro da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES.

Resumo da Dissertação apresentada ao PROEE/UFS como parte dos requisitos necessários para a obtenção do grau de Mestre (Me.)

PLANEJAMENTO DE MOVIMENTO AGRESSIVO PARA UM SISTEMA DE QUADRICÓPTERO COM CARGA SUSPENSA BASEADO EM RRT

JEFFERSON DE LIMA SILVEIRA JÚNIOR

Março/2020

Orientador: Sidney Nascimento Givigi Júnior

Co-orientador: Eduardo Oliveira Freire

A utilização de Veículos Aéreos Não Tripulados (VANTs), também chamados de *drones*, tem crescido bastante. Isto se deve ao desenvolvimento da tecnologia, que permitiu uma maior facilidade de utilização, versatilidade de aplicações e redução dos custos de produção. Os setores favorecidos com o crescimento da comercialização de *drones* são vários e, entre eles, está o setor de transporte. Nesse setor, uma aplicação pouco explorada que pode trazer benefícios para a sociedade é adicionar a capacidade de transporte de cargas em *drones*. Isso pode ser feito através de um cabo com mecanismos de fixação no *drone* e na carga. Em contrapartida, a adição do cabo permite a oscilação da carga, que gera uma necessidade do desenvolvimento de arquiteturas de planejamento de movimento e controle capazes de lidar com o problema das oscilações de forma eficiente. Entre as abordagens utilizadas para resolver o problema, estão as aplicações de aprendizado por reforço e otimização quadrática para gerar trajetórias livres de obstáculos ótimas e realizáveis. Embora tais técnicas forneçam um planejamento de movimento ótimo, o sucesso da solução depende da restrição dos obstáculos a geometrias simples e convexas. Nesta dissertação, é proposto um planejador de movimento que gera trajetórias que respeitam as restrições dinâmicas e cinemáticas e que não restringe a geometria dos obstáculos. Para tal, é proposta uma variante do algoritmo *Rapidly-exploring Random Tree* (RRT), que é um algoritmo de planejamento aleatório. A partir dos resultados obtidos em ambientes com diferentes tipos de obstáculos, foi possível verificar que o algoritmo proposto é capaz de solucionar o problema de planejamento de movimento em diversas situações.

Palavras-chaves: planejamento de movimento, planejamento probabilístico, robótica aérea.

Abstract of Dissertation presented to PROEE/UFS as a partial fulfillment of the requirements for the degree of Master

AGGRESSIVE MOTION PLANNING FOR A QUADROTOR SYSTEM WITH SLUNG LOAD BASED ON RRT

JEFFERSON DE LIMA SILVEIRA JÚNIOR

February/2020

Advisor: Sidney Nascimento Givigi Júnior

Co-advisor: Eduardo Oliveira Freire

The application of Unmanned Aerial Vehicles (UAVs), also called drones, has been growing lately. That is due to advances in technology, which has made them more accessible, versatile, and has also reduced production costs. Drone technology has been used in many commercial applications, including the transport segment. In this segment, drones can bring benefits to society as load vehicles. For this purpose, a cable is used to suspend and transport the load with the drone. However, the cable allows oscillations in the load, which requires the development of motion planning and control architectures capable of handling the oscillation efficiently. Many algorithms can be used to solve such problems, including reinforcement learning or quadratic programming to generate feasible and optimal collision-free trajectories. Although these techniques provide optimal motion planning, the success of finding a solution depends on geometric constraints for simple and convex obstacles. This dissertation proposes a motion planner that generates feasible trajectories and does not constrain the obstacle geometry. To this end, a variant of the algorithm Rapidly-exploring Random Tree (RRT) is proposed, which is a randomized motion planning algorithm. Results show that the proposed algorithm is able to solve the motion planning problem in many situations.

Key-words: motion planning, probabilistic planning, aerial robotics.

Lista de ilustrações

Figura 1 – Ilustrações dos manifolds que surgem durante a modelagem do sistema quadricóptero-carga. 12
Figura 2 – Um quadricóptero com carga suspensa via cabo ilustrado com dois sistemas de coordenadas, um inercial e um fixo no quadricóptero. . . . 13
Figura 3 – Estrutura do controlador geométrico para rastrear a posição da carga. . 17
Figura 4 – Geração do sistema de coordenadas que constituem a matriz de rotação $\boldsymbol{R}_c$. O vetor unitário $\boldsymbol{b}_{3_c}$ é calculado em (3.23), $\boldsymbol{b}_{1_d}$ é escolhido arbitrariamente não sendo paralelo a $\boldsymbol{b}_{3_c}$ e está relacionado com a guinada, que define a direção que quadricóptero estará "olhando". Com estes dois vetores, é possível obter $\boldsymbol{b}_{1_c}$ e $\boldsymbol{b}_{2_c}$. 20
Figura 5 – Crescimento da RRT após 5 expansões de nós. 24
Figura 6 – Crescimento da RRT clássica em três momentos: (a) a RRT começa a se expandir em direção às regiões de Voronoi maiores; (b) a RRT continua se expandindo com ramos menores; (c) a RRT amostra um ponto próximo do destino e retorna uma trajetória. Os três círculos representam obstáculos. 25
Figura 7 – Exemplos de RRT: (a) de um pêndulo com trajetórias apresentadas no no espaço de estados e (b) de um carro com trajetórias apresentadas no espaço de trabalho. Ambos implementados por Shkolnik, Walter e Tedrake (2009). 26
Figura 8 – Quadricóptero planar com carga suspensa. 28
Figura 9 – Trajetórias executadas pelo sistema quadricóptero carga com o controlador geométrico: (a) trajetória circular de raio 1.5 m e período 5 s e (b) trajetória helicoidal de raio 1 m, período 2π s e velocidade ascendente de 1 m/s. 31
Figura 10 – Trajetórias circulares com períodos (T) diferentes: (a) $T = 5\ s$ (b) $T = 4\ s$ e (c) $T = 3\ s$. Todas as trajetórias foram executadas por $3T\ s$. 33
Figura 11 – Ilustração dos efeitos de métricas de vizinhança diferentes na RRT utilizando pêndulos. Apesar do pêndulo verde estar mais próximo, é mais fácil para o pêndulo azul atingir o estado representado pelo pêndulo vermelho. 38
Figura 12 – Ambiente 1, sem obstáculos. 43
Figura 13 – Ambiente 2: (a) vista superior e (b) vista panorâmica. Este ambiente possui obstáculos que exigem movimentação no plano xy. 44

Figura 14 – Ambiente 3: (a) vista lateral e (b) vista panorâmica. Este ambiente possui obstáculos que exigem movimentação no plano zy. 44
Figura 15 – Ambiente 4: (a) vista lateral e (b) vista panorâmica. Este ambiente possui obstáculos cuja abertura é 85% do comprimento do sistema quadricóptero-carga: 45
Figura 16 – RRT, projetada no espaço tridimensional, gerada no Ambiente 1: (a) vista superior e (b) vista lateral. 46
Figura 17 – Trajetórias geradas pela RRT proposta para o Ambiente 1: (a) contém 10 trajetórias entre as 100 obtidas e (b) contém a trajetória mais curta e a mais longa. 46
Figura 18 – Esforço de controle dos motores referente à trajetória mais curta obtida pela RRT no Ambiente 1. 47
Figura 19 – Histograma do tempo de solução para o Ambiente 1. 48
Figura 20 – RRT, projetada no espaço tridimensional, gerada no Ambiente 2: (a) vista superior e (b) vista lateral. 49
Figura 21 – Trajetórias, contendo a mais longa e a mais curta, geradas pela RRT proposta no Ambiente 2: (a) vista lateral e (b) vista superior. 49
Figura 22 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 2. 50
Figura 23 – Velocidades dos motores executadas na solução da Figura 22a. 50
Figura 24 – Histograma do tempo de solução para o Ambiente 2. 51
Figura 25 – RRT, projetada no espaço tridimensional, gerada no Ambiente 3: (a) vista lateral e (b) vista superior. 52
Figura 26 – Trajetórias, contendo a mais longa e a mais curta, geradas pela RRT proposta no Ambiente 3: (a) vista lateral e (b) vista superior. 52
Figura 27 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 2. 53
Figura 28 – Velocidades dos motores executadas na solução da Figura 27a. 53
Figura 29 – Histograma do tempo de solução para o Ambiente 3. 54
Figura 30 – RRT, projetada no espaço tridimensional, gerada no Ambiente 4: (a) vista superior e (b) vista lateral. 55
Figura 31 – Trajetórias geradas pela RRT proposta no Ambiente 4: (a) vista superior e (b) vista lateral. 55
Figura 32 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 4. 56
Figura 33 – Esforço de controle dos motores de da solução apresentada na Figura 32b. 56
Figura 34 – Histograma do tempo de solução para o Ambiente 4. 57
Figura 35 – Ambiente com obstáculo não convexo: (a) vista lateral e (b) vista panorâmica. 58
Figura 36 – Trajetória gerada pela RRT proposta no ambiente com obstáculo não convexo: (a) projeção da RRT no espaço tridimensional e (b) movimentação resultante. 59

Lista de tabelas

Tabela 1 – Símbolos e descrições relacionados ao modelo do quadricóptero com carga suspensa. 14
Tabela 2 – Parâmetros do quadricóptero com carga suspensa. 15

Tabela 3 – Parâmetros do quadricóptero com carga suspensa. 31
Tabela 4 – Ganhos do controlador geométrico. 32

Tabela 5 – Ganhos do controlador geométrico. 43
Tabela 6 – Dados das soluções no Ambiente 1. 48
Tabela 7 – Dados das soluções com exclusão de nós no Ambiente 2. 51
Tabela 8 – Dados das soluções com exclusão de nós no Ambiente 3. 54
Tabela 9 – Dados das soluções com exclusão de nós no Ambiente 4. 57
Tabela 10 –Comparação das taxas de sucesso com e sem exclusão de nós. 61

Sumário

Lista de ilustrações . . . . . **VII**

Lista de tabelas . . . . . **IX**

1 Introdução **1**
1.1 Objetivos 3
1.2 Estrutura do trabalho 4

2 Revisão Bibliográfica **5**
2.1 Controle e planejamento de movimento do sistema quadricóptero-carga . . 5
2.2 Planejamento de movimento dinâmico com RRT 8

3 Fundamentação Teórica **11**
3.1 Breve abordagem sobre os *manifolds* S^2 e $SO(3)$ 11
3.2 Modelagem do sistema 12
3.3 Projeto dos Controladores Geométricos 16
3.3.1 Controlador de orientação do quadricóptero 18
3.3.2 Controlador de orientação da carga 19
3.3.3 Controlador de posição da carga 21
3.4 Estimação das derivadas 22
3.5 Árvore Aleatória de Exploração Rápida (RRT) 23

4 Desenvolvimento do Trabalho e Abordagem Proposta **28**
4.1 Estudo inicial do sistema quadricóptero-carga planar 28
4.2 Implementação do Sistema Quadricóptero-Carga Controlado 30
4.3 Planejamento de Trajetórias Baseado no Controlador Geométrico 34
4.4 Definição do espaço de busca da RRT 35
4.5 Polarização da RRT em direção ao destino 36
4.6 Obtenção do vizinho mais próximo 37
4.7 Expansão da árvore 39
4.8 Detecção de colisão 40
4.9 Planejador de movimento completo 41

5 Resultados e Discussão **42**
5.1 Ambientes de teste do algoritmo proposto 43
5.1.1 Ambiente 1 43
5.1.2 Ambiente 2 43

5.1.3 Ambiente 3 44
5.1.4 Ambiente 4 45
5.2 Experimentos com o Ambiente 1 45
5.3 Experimentos com o Ambiente 2 48
5.4 Experimentos com o Ambiente 3 51
5.5 Experimentos com o Ambiente 4 54
5.6 Teste em um ambiente com um obstáculo não convexo 57
5.7 Desafios observados com os experimentos 59

6 Conclusões 62

Referências 64

Apêndices 69

APÊNDICE A Fundamentos matemáticos sobre os *manifolds* S^2 e $SO(3)$ . . 70

APÊNDICE B Obtenção das equações de movimento 73

APÊNDICE C Funções de erro de rastreamento 78

1 Introdução

Nestes últimos anos, a utilização de Veículos Aéreos Não Tripulados (VANTs), também chamados de *drones*, tem crescido bastante. O mercado que continha prioritariamente os segmentos militares e hobistas, agora também inclui o segmento comercial (MOLINA; CAMPOS, 2018). Uma pesquisa de mercado feita por Moe et al. (2017) mostra que, entre três segmentos, o que tem a maior taxa de crescimento é o comercial, com estimativas de que deve alcançar 500 milhões de dólares de valor de mercado em 2020.

O crescimento da comercialização de *drones* deve acelerar ainda mais com a evolução da tecnologia, da regulamentação e do surgimento de novas aplicações. Mazur, Wisniewski e McMillan (2016) afirmam que os setores que serão favorecidos com o avanço do mercado e da tecnologia de *drones* são os de infraestrutura, transporte, segurança, entretenimento, telecomunicação, agricultura, seguros e mineração. Entre as diversas aplicações de *drones* nestes segmentos, podem ser citadas aquelas relacionadas à saúde, no transporte de sangue (AMUKELE et al., 2017) ou mapeamento de doenças (HARDY et al., 2017); aplicações ecológicas no mapeamento da fauna (HODGSON; PEEL; KELLY, 2017) ou da flora (SANKEY et al., 2017); aplicações comerciais com serviços de entrega, como o projeto *Amazon Prime Air* (PANDIT; POOJARI, 2014), entre outras.

A crescente demanda de *drones* e a procura de novas aplicações na indústria têm fomentado o desenvolvimento de novas tecnologias e de pesquisas relacionadas a expandir a aplicabilidade de veículos aéreos. Um exemplo disso são pesquisas que envolvem o acoplamento de garras para que um *drone* possa pegar e manipular objetos (MELLINGER et al., 2011; CANO et al., 2013). Considerando que uma parcela das possíveis aplicações de *drones* na indústria envolvem o transporte ou manipulação de objetos, adicionar mecanismos que permitam tais atividades é uma opção que vem sendo investigada.

De acordo com Kumar e Michael (2012), os quadricópteros, que são *drones* que possuem quatro motores, representam a melhor configuração considerando a capacidade de carregar cargas pequenas. Contudo, a adição de estruturas rígidas para manipulação de objetos em um quadricóptero aumenta bastante o peso do sistema. Como resultado, o tempo de voo é reduzido e a capacidade de realização de manobras é afetada com o aumento da inércia. Uma opção mais simples é anexar uma corda ao *drone* com um mecanismo de conexão à carga na outra ponta. Por ser mais leve, o *drone* não perde agilidade. A carga útil que pode ser transportada é maior e, além disso, é possível transportar objetos por mais tempo. Por outro lado, a adição da corda permite oscilação da carga e impossibilita a realização de tarefas que requerem cuidado especial, como o transporte de órgãos.

Quando a carga não necessita de cuidado especial, a opção de transporte via cabo costuma ser mais adequada, pois as oscilações podem ser tratadas pelo sistema de controle. No entanto, atuar no sistema correspondente ao quadricóptero com carga suspensa via cabo é uma tarefa muito mais complexa, pois o controlador deixa de ter que lidar com um sistema de apenas um corpo rígido e passa a lidar com um sistema de dois corpos acoplados, em que a carga suspensa pelo cabo age de forma semelhante a um pêndulo esférico. Sob uma perspectiva acadêmica, a necessidade de uma arquitetura de controle capaz de lidar com o problema de forma eficiente é um desafio que vêm sendo bastante investigado na comunidade acadêmica devido à complexidade do sistema com dinâmica acoplada.

O problema de transportar uma carga suspensa via cabo é um tema relativamente antigo na literatura por ser algo que já vem sendo pesquisado em helicópteros convencionais. Lucassen e Sterk (1965), por exemplo, fizeram a análise de estabilidade de um helicóptero com carga suspensa ainda em 1965. Entretanto, uma das primeiras abordagens encontradas na literatura utilizando um quadricóptero para transportar uma carga suspensa via cabo é realizada por Palunko, Fierro e Cruz (2012). Nesse trabalho, há a preocupação de gerar uma trajetória para o quadricóptero que não acarrete em oscilações na carga. No ano seguinte, Faust et al. (2013) propõem a aplicação de aprendizado por reforço para o planejamento de movimento de um quadricóptero também com o objetivo de reduzir as oscilações da carga.

Nas abordagens citadas, o objetivo de suprimir a oscilação da carga parece ser vantajoso em um primeiro momento. Porém, o sistema de controle precisa gastar energia extra para eliminar um modo natural do sistema, que é oscilar, além de que é possível se aproveitar do movimento natural da carga para executar trajetórias mais otimizadas. Neste sentido, Tang, Sreenath e Kumar (2014) apresentam uma abordagem de controle que permite manobras agressivas da carga para executar uma tarefa. O trabalho realizado por eles foi inspirado em um video de um piloto de helicóptero bastante habilidoso colhendo árvores de natal e depositando em um caminhão enquanto se aproveita da oscilação da carga para otimizar seu tempo de trabalho[1]. Este trabalho foi possibilitado devido ao controlador não linear proposto por Sreenath, Lee e Kumar (2013) que mantém a estabilidade do sistema mesmo distante do ponto de operação.

Considerando a possibilidade de movimentações mais agressivas do sistema, outras abordagens também são apresentadas. Crousaz, Farshidian e Buchli (2014) apresentam um algoritmo de aprendizado iterativo capaz de fazer o quadricóptero com carga suspensa atravessar uma janela cuja altura é menor que o comprimento do cabo. Esta situação exige ao sistema de planejamento e controle a execução uma trajetória que se aproveita da dinâmica do sistema para evitar colisões com a janela.

[1] Este vídeo pode ser encontrado em https://www.youtube.com/watch?v=SPO9pVwoxVg.

O problema de atravessar uma janela curta também é abordado por Tang e Kumar (2015), mas eles utilizam um gerador de trajetória baseado em programação quadrática e um controlador geométrico não linear para executar a trajetória. Ainda atacando o problema de atravessar uma janela, Geisert e Mansard (2016) também apresentam uma abordagem de programação quadrática, mas também solucionam o problema utilizando controle preditivo para a solução *online* da trajetória.

Todas as técnicas citadas para resolver o problema da janela envolvem a utilização de ferramentas matemáticas para adicionar restrições no sistema. Sejam elas para evitar obstáculos ou limitar as velocidades e acelerações do modelo. No entanto, as etapas de otimização partem do princípio que o problema a ser otimizado é convexo. Esta consideração pode resultar em dificuldade de convergência para tarefas mais complexas e limita a construção do mapa, pois os obstáculos devem ser convexos e possuir estruturas geométricas simples para o equacionamento das restrições dos algoritmos de otimização.

Nesse contexto, este trabalho propõe resolver o problema de atravessar uma janela curta considerando o sistema do quadricóptero carregando uma carga suspensa, ou sistema quadricóptero-carga para abreviar, de uma maneira que não seja necessário limitar o formato dos obstáculos para que o problema possa ser resolvido. Adicionalmente, deve-se levar em consideração que o espaço de busca do problema em questão é de alta dimensão e a solução precisa respeitar as restrições cinemáticas e dinâmicas para que se encontre uma trajetória factível.

A técnica de planejamento de movimento que mais se encaixa no problema é a *Rapidly-exploring Random Tree* (RRT). A RRT é uma técnica probabilística que utiliza amostragens aleatórias para buscar em espaços com grandes dimensões, permite facilmente a adição de restrições dinâmicas e não necessita de funções de custo para resolver o problema. A não dependência de funções de custo é uma grande vantagem, pois evita problemas como ocorrência de mínimos locais, principalmente quando são utilizados ambientes complexos que podem conter obstáculos convexos. Ao passo que a utilização da RRT permite encontrar soluções viáveis através de uma rápida busca no espaço, não há garantias de que a solução é ótima.

1.1 Objetivos

O objetivo geral deste trabalho é propor um algoritmo de planejamento de movimento baseado na RRT para resolver problemas em que manobras agressivas são necessárias para transpor os obstáculos.

São objetivos específicos deste trabalho:

- Realizar uma ampla revisão bibliográfica acerca das abordagens de planejamento e controle de quadricópteros no transporte de carga suspensa;
- Investigar implementações baseadas em RRT para sistemas com restrições cinemáticas e dinâmicas;
- Simular o modelo matemático do sistema quadricóptero-carga;
- Implementar um controlador para solucionar o problema de rastreamento de trajetória da carga considerando manobras agressivas;
- Propor e implementar um planejador de movimentos baseado na RRT utilizando o controlador implementado;
- Realizar experimentos simulados e investigar a capacidade da abordagem proposta na solução do problema da janela curta.

1.2 Estrutura do trabalho

Esta dissertação está estruturada da seguinte maneira: no Capítulo 2, é apresentada uma revisão acerca das abordagens de geração de trajetória e do sistema de controle para quadricópteros com carga suspensa, bem como uma breve revisão sobre RRT; no Capítulo 3, é apresentada a fundamentação teórica referente a implementação do controlador e da RRT; no Capítulo 4 é apresentada a abordagem proposta; no Capítulo 5, os resultados são apresentados e discutidos; e, por fim, a conclusão é apresentada no Capítulo 6.

2 Revisão Bibliográfica

Neste capítulo, são apresentadas dois temas de revisão bibliográfica: uma referente ao controle e planejamento de movimento aplicados no sistema quadricóptero carga; e a segunda referente ao planejamento de movimento de sistemas dinâmicos utilizando o algoritmo RRT.

2.1 Controle e planejamento de movimento do sistema quadricóptero-carga

Quando se trata da utilização de quadricópteros para o transporte de carga, diversas abordagens de controle são utilizadas para reduzir as oscilações que normalmente ocorrem durante a execução de uma trajetória. Palunko, Cruz e Fierro (2012) apresentam uma discussão sobre as possíveis abordagens para reduzir as oscilações da carga em dois processos: um controlador adaptativo para lidar com possíveis problemas de deslocamento do centro de massa do sistema e um gerador de trajetórias ótimo para manobras livres de oscilações baseado em programação dinâmica para ser utilizado no controlador.

Um método de aprendizado por reforço é apresentado por Faust et al. (2013) para gerar trajetórias com oscilações residuais mínimas para um quadricóptero com carga suspensa. O aprendizado se dá por um algoritmo chamado *Approximate Value Iteration* (AVI). As etapas de aprendizado e de geração de trajetória são executadas em fases distintas. A primeira etapa envolve o aprendizado dos parâmetros de uma função utilizando um modelo simplificado do sistema. A principal vantagem desta abordagem é que o aprendizado ocorre sem o conhecimento do estado inicial e dos parâmetros do modelo, o que significa que o aprendizado só precisa ser feito uma vez e pode ser aplicado em diferentes condições iniciais. Uma vez que a função é obtida, a segunda etapa envolve gerar a trajetória que pode assumir diferentes posições iniciais e finais do sistema. Outra proposta de aprendizado por reforço utilizando mínimos quadrados iterativo é feita por Palunko et al. (2013).

Também com o objetivo de cancelar as oscilações do sistema, Sadr, Moosavian e Zarafshan (2014) propõem um controlador utilizando a técnica chamada *input shaping*. Nesta abordagem, impulsos são aplicados em tempos específicos no sistema de forma a cancelar as oscilações. Esta técnica é implementada a partir da convolução do comando de referência com uma sequência de impulsos que devem ocorrer em momentos específicos, a depender da frequência natural do sistema e do fator de amortecimento. Os instantes de ocorrência dos impulsos são escolhidos de forma que as oscilações do sistema sofram

interferências destrutivas, que resultam na redução das oscilações. Adicionalmente, o controle da posição e orientação é implementado com uma compensação dos termos não lineares baseando-se no conhecimento dos parâmetros do sistema.

Um sistema de controle adaptativo é implementado por Feng et al. (2015). Neste trabalho, é apresentado um método de controle que incorpora possíveis perturbações externas e também tem o objetivo de cancelar as oscilações geradas durante a execução de uma trajetória desejada.

Outra abordagem de controle adaptativo é discutida em Mo, Geng e Lu (2016). Neste artigo, o modelo de um quadricóptero é linearizado e desacoplado no ponto de operação em quatro subsistemas. Assim, um método melhorado do controle adaptativo L1 é aplicado a dois subsistemas denominados longitudinal e lateral e um controlador linear ativo de rejeição de perturbação (LADRC) é aplicado nos outros dois subsistemas (vertical e de orientação) com o objetivo de reduzir a oscilação da carga. Adicionalmente, é utilizado um algoritmo genético para otimizar a escolha dos parâmetros dos controladores.

Um controle de modos deslizantes é proposto por Kui et al. (2017) com o intuito de reduzir perturbações externas causadas pela oscilação da própria carga. Os controladores são divididos em dois grupos: os projetados para controlar a posição e os projetados para controlar a orientação. Então, são apresentados resultados de simulações validando a robustez às perturbações e validando o bom desempenho de rastreamento de posição do quadricóptero.

Crousaz, Farshidian e Buchli (2014) propõem uma abordagem de aprendizado por reforço, que se diferencia pelo fato de que o objetivo do controlador não é apenas cancelar as oscilações do sistema. O algoritmo de aprendizado por reforço é utilizado para fazer o quadricóptero com carga suspensa realizar movimentos agressivos permitindo a execução de tarefas complexas, como atravessar uma janela cuja altura é menor que o próprio quadricóptero com a carga. Para tal, utiliza-se um modelo híbrido, que consiste em permitir o chaveamento da dinâmica considerando a queda livre da carga a depender se o cabo está ou não tracionado. O algoritmo PI^2-01 é utilizado para o aprendizado que minimiza uma função de custo que deve ser cuidadosamente projetada para solucionar o problema desejado. Além disso, o algoritmo pode divergir ou violar as restrições de atuação se a função de custo for muito diferente de uma função convexa.

Uma abordagem utilizando um controlador preditivo é apresentada por Sun et al. (2018). Neste trabalho, além da supressão da oscilação da carga, é resolvido o problema de desvio de obstáculos. Para tal, o modelo do quadricóptero com carga suspensa é linearizado e discretizado. Além disso, é assumido que o quadricóptero possui um bom desempenho no rastreamento de trajetórias, portanto, o modelo dinâmico pode ser substituído pelo modelo cinemático com o intuito de simplificar e melhorar a eficiência do algoritmo. Então, é aplicado o algoritmo de controle preditivo de forma que a função de custo depende de três

fatores: da distância do sistema para a referência, dos esforços de controle e das distâncias para os obstáculos. Por fim, são apresentados resultados de simulações mostrando que o quadricóptero foi capaz de desviar de obstáculos e suprimir oscilações da carga ao mesmo tempo.

No trabalho apresentado por Sreenath, Lee e Kumar (2013), são propostos controladores não lineares para rastrear a posição da carga. A principal contribuição do trabalho é que o controlador é capaz de rastrear movimentos agressivos permitindo o rastreamento de estados distantes do ponto de equilíbrio. Esta abordagem consegue explorar bastante a agilidade do quadricóptero permitindo manobras mais agressivas. Por este motivo, o cabo que prende a carga ao quadricóptero pode perder tensão a depender da manobra. Então, é levantado um modelo dinâmico híbrido cujo chaveamento é função da tensão no cabo. A trajetória utilizada pelo controlador deve ser suave, conter a posição desejada da carga e a guinada do quadricóptero em cada instante de tempo e suas primeiras e segundas derivadas. Deste modo, o controlador é implementado de forma recursiva, também conhecido como *backstepping*, e encontra ações intermediárias que estabilizam os subsistemas, como a posição da carga, orientação da carga e orientação do quadricóptero com o objetivo de rastrear a trajetória desejada.

Com o controlador proposto por Sreenath, Lee e Kumar (2013), Tang e Kumar (2015) propõem um gerador de trajetórias para resolver diversos problemas, inclusive o problema da janela curta utilizando movimentações agressivas.

Recentemente, mais um trabalho enfatiza as vantagens de se utilizar um controle agressivo. Tang, Wüest e Kumar (2018) mostam que o sistema desenvolvido permite oscilações com ângulos de até 50° da carga. As vantagens citadas no artigo partem do princípio de que suprimir as oscilações naturais do sistema é uma abordagem sub-ótima, pois ignora a capacidade de manobras que o sistema possui ao permitir que o mesmo oscile de forma controlada. O sistema de controle é baseado na teoria de *flat outputs* e ele é aplicado de forma que a posição da carga faz parte do sistema realimentado, em contrapartida com outros artigos que controlam a posição do quadricóptero. Desta forma, é possível planejar e controlar a trajetória da carga. Além disso, outra importante contribuição deste trabalho é o desenvolvimento do estimador baseado em visão computacional, anexado na parte inferior do quadricóptero, utilizado para obter informações sobre o estado da carga.

Geisert e Mansard (2016) utilizam uma técnica de otimização que envolve discretizar o problema e dividir a trajetória em diversos pedaços e integrar a equação diferencial nos intervalos divididos. Desta forma, a otimização pode ocorrer de forma mais rápida e permite que a etapa de otimização possa ser executada de maneira *online* em um controlador preditivo. Um dos problemas utilizados para avaliar a abordagem proposta é o problema

da janela curta. Os resultados mostram que o problema de otimização é capaz de encontrar soluções, porém, houve problemas na etapa de adaptação da função de custo do sistema completo no controlador preditivo.

Os trabalhos realizados por Sreenath, Lee e Kumar (2013), Crousaz, Farshidian e Buchli (2014), Tang, Wüest e Kumar (2018) e Geisert e Mansard (2016) são semelhantes no ponto de vista de que todos desenvolveram sistemas de controle que permitem grandes oscilações da carga e uma maior manobrabilidade do quadricóptero. Porém, se diferenciam principalmente no fato de que Sreenath, Lee e Kumar (2013) e Crousaz, Farshidian e Buchli (2014) permitem que o cabo não esteja tracionado enquanto Tang, Wüest e Kumar (2018) utilizam uma ferramenta matemática para alterar os intervalos da trajetória obtida pelo sistema de otimização de forma a garantir que o cabo esteja sempre tracionado e Geisert e Mansard (2016) consideram uma barra rígida no lugar do cabo. Além disso, Crousaz, Farshidian e Buchli (2014) utilizam aprendizado por reforço enquanto Sreenath, Lee e Kumar (2013), Tang, Wüest e Kumar (2018) e Geisert e Mansard (2016) utilizam programação quadrática para otimizar o problema e um controlador geométrico para rastrear a trajetória.

Diferentemente de todas as abordagens citadas, Zeng, Kotaru e Sreenath (2019) propõem utilizar uma polia atuada para controlar o comprimento do cabo. Assim, eles resolvem o problema da janela curta alterando o comprimento do cabo até que o sistema seja capaz de atravessar a janela sem a necessidade de movimentos agressivos. Outro trabalho interessante é o de Goodarzi, Lee e Lee (2014), em que o modelo do sistema é levantado considerando que o cabo é flexível. A flexibilidade do cabo é modelada como diversos elos fixos conectados em série. Um controlador geométrico é utilizado para estabilizar o sistema. Mais um artigo que se diferencia envolve o estudo apenas do levantamento da carga. Neste trabalho, é considerado que a massa da carga é desconhecida (CRUZ; FIERRO, 2014) e também é utilizado um controlador geométrico para atuar no sistema.

2.2 Planejamento de movimento dinâmico com RRT

O algoritmo RRT baseia-se em uma estrutura de dados aleatória que cresce de forma semelhante a uma árvore. O algoritmo foi proposto para o planejamento de trajetórias que contêm restrições holonômicas. De acordo com LaValle (1998), criador da RRT, seu algoritmo possui o menor número possível de heurísticas e parâmetros. Isso leva a um melhor desempenho e comportamento consistente do algoritmo. Outro aspecto importante do algoritmo é que ele pode ser aplicado diretamente no planejamento de trajetórias com restrições dinâmicas e holonômicas. Deste modo, LaValle apresenta diversos exemplos do planejamento de trajetória de sistemas com restrições holonômicas.

Outra característica importante da RRT é que a generalização do algoritmo para altas dimensões costuma ser direta. No entanto, a busca de uma solução em um espaço de dimensão elevada pode demorar para ser encontrada. Pensando neste problema, Shkolnik e Tedrake (2009) utilizam a RRT para resolver problemas com mais de 1000 dimensões projetando o espaço de configuração no espaço de trabalho. Assim, a busca ocorre no espaço de trabalho, que costuma ser de dimensão bem menor, mas a árvore cresce no espaço de configuração reduzindo bastante o espaço de busca da solução e, consequentemente, acelerando o tempo de solução do algoritmo.

Devido à simplicidade do algoritmo e a facilidade de adicionar novas heurísticas, diversas variantes do algoritmo foram desenvolvidas ao longo dos anos. Karaman e Frazzoli (2011) discutem algumas abordagens probabilísticas e, entre elas, o algoritmo RRT*. O RRT* herda da RRT o fato de que o algoritmo é probabilisticamente completo, o que significa que a probabilidade de encontrar uma solução se aproxima de 1 com o passar do tempo de execução do algoritmo. Além disso, ele incorpora uma etapa de otimização da trajetória, o que significa que com o passar do tempo, a solução se aproxima assintoticamente da ótima.

O algoritmo RRT* é utilizado por Karaman et al. (2011) no desenvolvimento de um planejador de trajetórias que rapidamente encontra um caminho factível e, com mais tempo de processamento, otimiza a trajetória durante sua execução.

Os algoritmos RRT e RRT* foram definidos genéricos o suficiente para não precisarem definir a função que atua no sistema. Esta é uma tarefa que depende do sistema envolvido no problema. Considerando esta questão, Perez et al. (2012) sugerem o LQR-RRT* que envolve linearizar a dinâmica do sistema em um ponto de operação e aplicar o regulador linear quadrático (LQR) para atuar no sistema. Assim, a função de custo utilizada na etapa de otimização do LQR-RRT* provém do LQR. Esta é uma abordagem bastante interessante porque não só é definida uma função para atuar no sistema como esta função fornece o custo necessário para otimizar a trajetória. A desvantagem é que necessita da linearização do sistema, o que pode interferir na execução de manobras que desviam bastante do ponto de operação.

A RRT tem uma tendência de gerar diversos ramos nas árvores durante sua execução e muitas vezes estes ramos não crescem na direção da solução desejada. Com o objetivo de acelerar a solução do algoritmo, Shkolnik et al. (2011) discretizam as possíveis ações em um robô e encontram o conjunto de estados alcançáveis. Com isso, é possível rejeitar os estados que não contribuem na solução do problema, evitando a criação de diversos ramos indesejáveis na RRT. Esta abordagem foi utilizada para planejar o movimento de um robô com 12 atuadores e 22 graus de liberdade. No entanto, a busca é feita em um modelo simplificado e planar com apenas 5 elos, resultando em um espaço de configuração com 7 dimensões e 4 atuadores.

Uma abordagem de planejamento de movimento para sistemas *differentially flat* é proposta por Bascetta, Arrieta e Prandini (2017). Neste trabalho, o autor explora o fato de que sistemas *differentially flat* possuem um conjunto de variáveis, chamadas de *flat outputs*, que são definidas de forma que todos os estados e ações podem ser determinados a partir delas e suas derivadas sem a necessidade de integração. Assim, é gerado um banco de dados de possíveis trajetórias que respeitam as restrições dadas pelas *flat outputs* e então a RRT é construida a partir destas trajetórias. Dessa maneira, a RRT pode ser construída eficientemente, sem integrar o sistema, e com trajetórias factíveis.

Um exemplo de abordagem utilizando RRT para a solução de um problema com quadricópteros pode ser encontrado em Webb e Berg (2013). Neste trabalho, o autor propõe uma modificação da RRT* utilizando uma função de custo que penaliza os esforços de controle e o tempo de execução da trajetória e são apresentadas soluções de tempo final fixo e de tempo de chegada ótimo. Então, é apresentada uma solução em forma fechada para o caso de sistemas nilpotentes e a RRT proposta é utilizada para resolver o problema de um quadricóptero atravessando janelas.

3 Fundamentação Teórica

Neste capitulo, é apresentada uma breve introdução a respeito dos espaços em que as configurações do sistema quadricóptero-carga evoluem. Em seguida, é apresentado o desenvolvimento matemático do modelo do sistema quadricóptero-carga a partir do Princípio de Mínima Ação de Hamilton. Após obter as equações de movimento, é apresentado um controlador com estabilidade quase global, que possui a capacidade de fazer o sistema recuperar-se de quase qualquer orientação. Por fim, é apresentado com mais detalhes o funcionamento do algoritmo RRT que serviu de base para o desenvolvimento do trabalho.

3.1 Breve abordagem sobre os *manifolds* S^2 e $SO(3)$

Nesta seção, são apresentadas ideias fundamentais para a modelagem do sistema e para a teoria de controle não linear geométrico utilizadas neste trabalho.

O sistema analisado neste trabalho corresponde a um acoplamento entre um quadricóptero, que é modelado como um corpo rígido, e um pêndulo esférico cuja massa é considerada pontual. Um quadricóptero no espaço é representado por sua posição e orientação. Enquanto a posição evolui no $\mathbb{R}^3$, a orientação evolui de uma maneira completamente diferente, apesar de também poder ser representada em três dimensões.

No caso do pêndulo, a posição da massa pontual pode ser representada no $\mathbb{R}^3$, mas a restrição causada pelo comprimento fixo do cabo, por ser rígido ou estar tracionado, faz com que o pêndulo evolua em um subespaço do $\mathbb{R}^3$ que pode ser parametrizado com dois ângulos apenas, mostrando que o sub-sistema do pêndulo na verdade evolui em um espaço bidimensional.

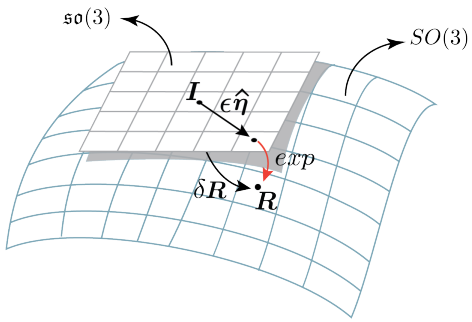
Entender as características e geometria de tais espaços auxilia não só na modelagem mas também na escolha de controladores mais robustos. O espaço em que a orientação do quadricóptero evolui é chamado de $SO(3)$ e o espaço em que o pêndulo esférico evolui é chamado de S^2. Ambos espaços são discutidos com mais detalhes nesta seção.

Antes de apresentar os detalhes, é necessário entender que espaço é esse e, para tal, é utilizada a definição apresentada por Spong, Hutchinson e Vidyasagar (2005): um *manifold* diferencial, ou *manifold* para abreviar, é um espaço topológico localmente difeomórfico[1] ao espaço Euclidiano $\mathbb{R}^m$. Ser localmente difeomórfico ao $\mathbb{R}^m$ significa que o *manifold*,

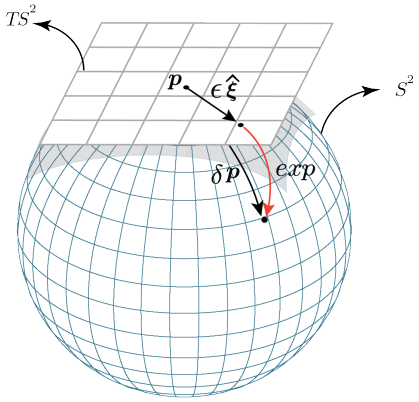
[1] Um difeomorfismo é uma função diferenciável cuja inversa existe e também é diferenciável. Tal função pode ser entendida como um isomorfismo de *manifolds*

localmente, assemelha-se a um espaço Euclidiano. Esta característica é importante porque pode-se trabalhar localmente com *manifolds* utilizando as ferramentas matemáticas já estabelecidas para o espaço Euclidiano.

Este fato é uma consequência de que, dado um *manifold* m-dimensional M, é possível associar a cada ponto $\boldsymbol{x} \in M$ um espaço tangente $T_{\boldsymbol{x}}M$ que é um espaço vetorial m-dimensional especificando o conjunto de possíveis derivadas direcionais em $\boldsymbol{x}$. Na Figura 1, são ilustrados dois *manifolds* com espaços tangentes em pontos específicos do espaço.



(a) Representação do manifold $SO(3)$ com o espaço tangente $\mathfrak{so}(3)$ na matriz identidade.



(b) Representação do manifold S^2 com seu espaço tangente em $\boldsymbol{p}$.

Figura 1 – Ilustrações dos manifolds que surgem durante a modelagem do sistema quadricóptero-carga.

A Figura 1 auxilia no entendimento das propriedades geométricas dos *manifolds* de interesse. Uma das principais características utilizadas neste trabalho está relacionada com a variação de uma configuração no *manifold*. Esta variação é utilizada para levantar o modelo matemático e para definir as funções de erro no controlador. Como os *manifolds* em questão são curvos, ilustrados nas Figuras 1a e 1b, é mais fácil encontrar tais variações no espaço tangente ao ponto desejado. Desta forma, por ser um espaço Euclidiano, a variação é dada por uma simples diferença. Então, o resultado pode ser mapeado de volta para a superfície do *manifold*, resultando em uma variação que respeita as restrições do sistema.

Os detalhes matemáticos para a obtenção das variações nestes *manifolds* estão apresentados no Apêndice A.

3.2 Modelagem do sistema

O modelo a ser obtido do sistema corresponde às equações de movimento que governam um quadricóptero carregando uma carga suspensa a partir de um cabo. Para isso, são feitas as seguintes considerações:

- A estrutura do quadricóptero é rígida e simétrica.
- Devido à simetria do sistema, a matriz de inércia J_Q, definida na Tabela 1, é considerada diagonal e constante.
- O centro de massa coincide com o ponto de contato entre o quadricóptero e a carga.
- O cabo anexado à carga é rígido e sem massa.
- Efeitos aerodinâmicos são desconsiderados.

Com a consideração do cabo rígido, a posição do quadricóptero e da carga estão relacionadas pela orientação e comprimento do cabo, este sendo constante e conhecido. Portanto, o sistema quadricóptero-carga é completamente representado sabendo a posição da carga $\boldsymbol{x}_L \in \mathbb{R}^3$, a orientação do quadricóptero $\boldsymbol{R} \in SO(3)$ e a orientação da carga $\boldsymbol{p} \in S^2$, resultando em 8 graus de liberdade.

O sistema quadricóptero-carga é ilustrado na Figura 2. Nesta imagem, são apresentados dois sistemas de coordenadas, o inercial $\mathcal{I}$ e o fixo ao corpo do quadricóptero $\mathcal{B}$. São também ilustradas as forças $F_{1..4}$ e momentos $M_{1..4}$ produzidos pelos atuadores. Todos os símbolos que independem de medidas do modelo do quadricóptero estão apresentados na Tabela 1 e os símbolos que dependem estão apresentados na Tabela 2.

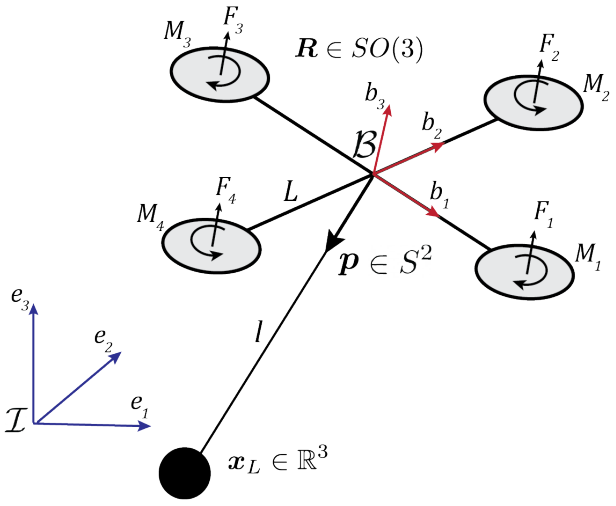


Figura 2 – Um quadricóptero com carga suspensa via cabo ilustrado com dois sistemas de coordenadas, um inercial e um fixo no quadricóptero.

As forças e momentos externos são produzidos pelos quatro atuadores anexados à sua estrutura e são funções da velocidade angular dos motores w_i de acordo com

$$F_i = k_F {w_i}^2, \tag{3.1}$$

$$M_i = k_M {w_i}^2, \tag{3.2}$$

Tabela 1 – Símbolos e descrições relacionados ao modelo do quadricóptero com carga suspensa.

Símbolo	**Descrição**
$\boldsymbol{x}_Q, \boldsymbol{x}_L \in \mathbb{R}^3$	Posição do quadricóptero e da carga, respectivamente
$\boldsymbol{p} \in S^2$	Vetor unitário do quadricóptero para a carga definido no espaço 2-esfera.
$\boldsymbol{v}_L \in \mathbb{R}^3$	Velocidade da carga
$g \in \mathbb{R}$	Aceleração da gravidade
$\boldsymbol{R} \in SO(3)$	Orientação do sistema de coordenadas fixo no quadricóptero com relação ao sistema de coordenadas inercial
$\boldsymbol{e}_3 \in \mathbb{R}^3$	Vetor unitário oposto ao vetor aceleração da gravidade
$f \in \mathbb{R}$	Magnitude da força de empuxo (*thrust*) do quadricóptero
$\boldsymbol{\omega} \in \mathbb{R}^3$	Velocidade angular da carga
$\boldsymbol{\Omega} \in \mathbb{R}^3$	Velocidade angular do quadricóptero no sistema de coordenadas fixo no corpo
$\boldsymbol{J}_Q \in \mathbb{R}^{3\times3}$	Matriz diagonal contendo os elementos de inércia I_{xx}, I_{yy}, and I_{zz}
$\boldsymbol{M} \in \mathbb{R}^3$	Momento vetorial do quadricóptero no sistema de coordenadas fixo no corpo

em que as constantes k_F e k_M são obtidas a partir de experimentos com o motor real. Então, as forças e momentos gerados por cada motor são combinados da seguinte maneira

$$\begin{bmatrix} f \\ \boldsymbol{M} \end{bmatrix} = \begin{bmatrix} k_F & k_F & k_F & k_F \\ 0 & k_F L & 0 & -k_F L \\ -k_F L & 0 & k_F L & 0 \\ k_M & -k_M & k_M & -k_M \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix}, \tag{3.3}$$

em que L é o comprimento do motor até o centro de massa do quadricóptero, f é a força resultante ortogonal ao corpo do quadricóptero, na mesma direção que $\boldsymbol{b}_3$, e $\boldsymbol{M}$ é o vetor momento angular resultante que age no sistema de coordenadas $\mathcal{B}$. Desta forma, f e $\boldsymbol{M}$ são considerados como as entradas do sistema.

Em seguida, as equações de movimento do sistema quadricóptero-carga são obtidas. Uma forma comum para modelar sistemas semelhantes se baseia em parametrizar a orientação do quadricóptero utilizando um sistema de coordenadas local, como os ângulos de Euler. Após parametrizar o sistema, as equações de Newton ou de Lagrange são utilizadas. No entanto, esta abordagem costuma produzir pontos de singularidade, equações extensas, complexas e cheias de funções trigonométricas que dificultam o projeto de sistemas de controle mais robustos.

Tabela 2 – Parâmetros do quadricóptero com carga suspensa.

Símbolo	**Descrição**
h	Altura
L	Comprimento do braço
l	Comprimento do cabo
m_Q	Massa do quadricóptero
m_L	Massa da carga
I_{xx}	Momento de inércia em X
I_{yy}	Momento de inércia em Y
I_{zz}	Momento de inércia em Z
w_{max}	Velocidade máxima do motor
w_{min}	Velocidade mínima do motor
k_F	Constante de força do moto
k_M	Constante de momento do motor

Para evitar tais problemas, a modelagem é abordada sem parametrizar a orientação do quadricóptero nem a orientação da carga. Esta abordagem é possível considerando a geometria do espaço em que tais orientações variam. Este método foi desenvolvido em vários sistemas que evoluem no $SO(3)$ e no S^2 por Lee, McClamroch e Leok (2005), Lee (2008) e Lee, Leok e McClamroch (2009). Uma abordagem semelhante aplicada para o sistema quadricóptero-carga é feita por Sreenath, Lee e Kumar (2013) e apresentada com bastante detalhes em Tang, Sreenath e Kumar (2014). O modelo resultante é livre de singularidades, permitindo o desenvolvimento de controladores com propriedades de convergência quase global.

A modelagem é baseada na obtenção das equações de movimento partindo do Princípio de Hamilton que afirma que:

> O movimento do sistema de um tempo t_1 para o tempo t_2 ocorre de forma que a integral de linha (chamada de integral de ação),
>
> $$S = \int_{t_1}^{t_2} \mathcal{L}\, dt, \tag{3.4}$$
>
> em que $\mathcal{L} = \mathcal{T} - \mathcal{U}$, possui valor estacionário para a trajetória de fato executada durante o movimento (GOLDSTEIN; POOLE; SAFKO, 2002, p. 34).

O Princípio de Hamilton é uma condição necessária e suficiente para as equações de Lagrange, em que $\mathcal{L} = \mathcal{T} - \mathcal{U}$ de forma que $\mathcal{T}$ é a energia cinética e $\mathcal{U}$ é a energia potencial. Em outras palavras, ambas as formulações, de Hamilton e de Lagrange, são equivalentes e é possível obter uma a partir da outra (GOLDSTEIN; POOLE; SAFKO, 2002).

O Princípio de Hamilton pode ser resumido ao dizer que o movimento ocorre de forma que a variação da integral de linha S para os tempos fixos t_1 e t_2 é zero:

$$\delta S = \int_{t_1}^{t_2} \delta\mathcal{L}\ dt = 0. \tag{3.5}$$

Para sistemas com forças ou momentos não conservativos, deve-se considerar a seguinte equação

$$\delta S = \int_{t_1}^{t_2} (\delta W + \delta\mathcal{L})\ dt = 0, \tag{3.6}$$

em que W representa o trabalho virtual resultante das forças geradas pelos motores.

A partir desta formulação, as equações de movimento do sistema podem ser obtidas ao igualar os termos multiplicativos das variações a zero. Os detalhes desse procedimento estão apresentados no Apêndice B e as equações de movimento resultantes são

$$\dot{\boldsymbol{x}}_L = \boldsymbol{v}_L, \tag{3.7}$$

$$(m_L + m_Q)(\dot{\boldsymbol{v}}_L + g\boldsymbol{e}_3) = (\boldsymbol{p} \cdot f\boldsymbol{R}\boldsymbol{e}_3 - m_Q l(\dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}}))\boldsymbol{p}, \tag{3.8}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{\omega} \times \boldsymbol{p}, \tag{3.9}$$

$$m_Q l\dot{\boldsymbol{\omega}} = -\boldsymbol{p} \times f\boldsymbol{R}\boldsymbol{e}_3, \tag{3.10}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\Omega}}, \tag{3.11}$$

$$\boldsymbol{J}_Q\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \boldsymbol{J}_Q\boldsymbol{\Omega} = \boldsymbol{M}, \tag{3.12}$$

em que as equações de (3.7) a (3.10) representam as dinâmicas da posição e orientação da carga e as equações (3.11) e (3.12) representam a dinâmica da orientação do quadricóptero. Além disso, o estado do sistema é dado por

$$\boldsymbol{\mathcal{X}} = [\boldsymbol{x}_L, \boldsymbol{v}_L, \boldsymbol{p}, \boldsymbol{\omega}, \boldsymbol{R}, \boldsymbol{\Omega}]. \tag{3.13}$$

3.3 Projeto dos Controladores Geométricos

A teoria de controle geométrico lida com o desenvolvimento de sistemas de controle para sistemas dinâmicos que evoluem em *manifolds* não lineares, que não podem ser identificados como espaços Euclidianos (LEE; LEOK; MCCLAMROCH, 2010). Nesta seção, são apresentados os controladores desenvolvidos por Sreenath, Lee e Kumar (2013)

para o sistema quadricóptero-carga baseados nos controladores de Lee, Leok e McClamroch (2010), que foram desenvolvidos com o intuito de estabilizar manobras complexas para um quadricóptero.

O sistema quadricóptero-carga possui apenas 4 graus de atuação sendo que a carga possui três graus de liberdade de posição, dois graus de liberdade de orientação e o quadricóptero possui mais três graus de liberdade de orientação. É impossível atingir rastreamento assintótico de todas as configurações com apenas 4 atuadores. Esta característica motivou Lee, Leok e McClamroch (2010) a desenvolver diversos modos de controle que podem ser combinados para gerar manobras mais complexas. São eles: um controlador de orientação, um de velocidade e um de posição.

O sistema de controle apresentado por Sreenath, Lee e Kumar (2013) possui uma característica recursiva de modo que são desenvolvidos vários subsistemas de controle, alguns baseados no trabalho de Lee, Leok e McClamroch (2010), que são interligados através de diversos ramos de realimentação com o objetivo de estabilizar a posição da carga e a guinada do quadricóptero. A estrutura do sistema de controle é apresentada na Figura 3, em que é possível ver o controlador de orientação diretamente conectado com o sistema quadricóptero-carga, que por sua vez está conectado no controlador de orientação da carga e, no último laço de realimentação, encontra-se o controlador de posição da carga.

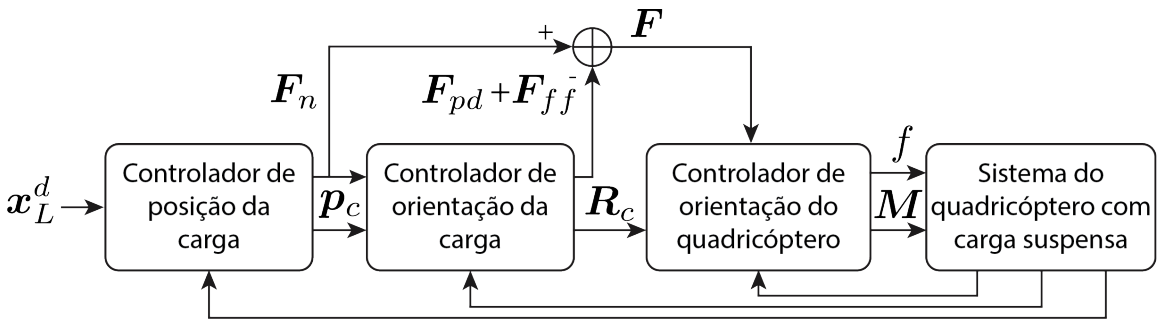


Figura 3 – Estrutura do controlador geométrico para rastrear a posição da carga.

Este sistema de controle garante rastreamento quase global das características do quadricóptero considerando que a região de atração de orientação cobre quase todo o espaço $SO(3)$. Como o controlador de posição contém em suas camadas de realimentação o controlador de orientação, o sistema controlado é capaz de executar manobras bastante agressivas sem perder estabilidade. Além disso, partindo do modelo apresentado na seção 3.2, o controlador é livre de parametrização, portanto, evita singularidades, ambiguidades e descontinuidades que surgem quando são utilizadas coordenadas locais, como ângulos de Euler.

Os controladores de orientação da carga e do quadricóptero atuam em seus respectivos *manifolds*. Por este motivo, são apresentadas as definições das funções de erro de rastreamento para o S^2 e o $SO(3)$ no Apêndice C. As seções seguintes são utilizadas para apresentar cada subsistema de controle.

Na apresentação dos controladores, é utilizada a notação com subscrito d significando valor desejado e subscrito c significando valor calculado. O valor desejado é referente à entrada e o valor calculado é referente à saída de cada subsistema de controle. Quando dois subsistemas são interligados, a saída de um será igual à entrada do outro, portanto, as variáveis desejadas e calculadas serão as mesmas.

3.3.1 Controlador de orientação do quadricóptero

O controlador de orientação do quadricóptero é projetado para fazer com que o quadricóptero possa rastrear uma trajetória de orientação $\boldsymbol{R}_d(t)$ diferenciável. O caminho para se obter a equação de controle é entender a dinâmica da função de erro definida em (C.13). Sua derivada no tempo é dada por

$$\dot{\boldsymbol{e}}_\Omega = \dot{\boldsymbol{\Omega}} + \hat{\boldsymbol{\Omega}}\boldsymbol{R}^T\boldsymbol{R}_d\boldsymbol{\Omega}_d - \boldsymbol{R}^T\boldsymbol{R}_d\dot{\boldsymbol{\Omega}}_d, \tag{3.14}$$

em que $\boldsymbol{\Omega}_d = (\boldsymbol{R}_d^T\dot{\boldsymbol{R}}_d)^\vee$.

Ao substituir (3.14) na dinâmica de orientação do quadricóptero (3.12) tem-se

$$\dot{\boldsymbol{e}}_\Omega = \boldsymbol{J}_Q^{-1}(-\boldsymbol{\Omega}\times\boldsymbol{J}_Q\boldsymbol{\Omega} + \boldsymbol{M}) + \hat{\boldsymbol{\Omega}}\boldsymbol{R}^T\boldsymbol{R}_d\boldsymbol{\Omega}_d - \boldsymbol{R}^T\boldsymbol{R}_d\dot{\boldsymbol{\Omega}}_d. \tag{3.15}$$

Então, Lee, Leok e McClamroch (2010) definem a entrada de controle $\boldsymbol{M}$ consistindo em termos proporcionais e derivativos acrescidos de termos para cancelar a não linearidade do sistema da seguinte forma

$$\boldsymbol{M} = -k_{\boldsymbol{R}}\boldsymbol{e}_{\boldsymbol{R}} - k_{\boldsymbol{\Omega}}\boldsymbol{e}_{\boldsymbol{\Omega}} + \boldsymbol{\Omega}\times\boldsymbol{J}_Q\boldsymbol{\Omega} - \boldsymbol{J}_Q(\hat{\boldsymbol{\Omega}}\boldsymbol{R}^T\boldsymbol{R}_d\boldsymbol{\Omega}_d - \boldsymbol{R}^T\boldsymbol{R}_c\dot{\boldsymbol{\Omega}}_d), \tag{3.16}$$

em que $\boldsymbol{k}_R$, $\boldsymbol{k}_\Omega$ são constantes positivas.

Ao substituir (3.16) na dinâmica do erro de velocidade angular (3.14), isso se reduz a

$$\boldsymbol{J}_Q\dot{\boldsymbol{e}}_{\boldsymbol{\Omega}} = -k_{\boldsymbol{R}}\boldsymbol{e}_{\boldsymbol{R}} - k_{\boldsymbol{\Omega}}\boldsymbol{e}_{\boldsymbol{\Omega}}, \tag{3.17}$$

que é uma equação diferencial simples cuja estabilidade é provada em Lee, Leok e McClamroch (2010) nas seguintes condições iniciais

$$\Psi_{\boldsymbol{R}}(\boldsymbol{R}(0), \boldsymbol{R}_d(0)) < 2, \tag{3.18}$$

$$||\boldsymbol{e}_{\boldsymbol{\Omega}}(0)||^2 < \frac{2}{\lambda_M(\boldsymbol{J}_Q)}k_{\boldsymbol{R}}(2 - \Psi_{\boldsymbol{R}}(\boldsymbol{R}(0), \boldsymbol{R}_d(0))), \tag{3.19}$$

em que $\lambda_M(\boldsymbol{J}_Q)$ representa o maior autovalor de $\boldsymbol{J}_Q$.

As equações (3.18) e (3.19) representam a região de atração do controlador de orientação. O primeiro termo requer que o erro de orientação seja menor que 180° em torno do eixo de rotação entre a orientação do quadricóptero inicial e a desejada. Com isso, região de atração então inclui praticamente todas as orientações iniciais. O segundo termo requer que o erro de velocidade angular seja menor que uma relação envolvendo a constante de ganho e o erro de orientação inicial. Como a região de atração praticamente cobre o $SO(3)$, este controlador é referido como contendo propriedades quase globais de atração com o ponto de equilíbrio dentro da região de atração sendo $(\boldsymbol{e_R}, \boldsymbol{e_\Omega}) = (\mathbf{0}, \mathbf{0})$ e exponencialmente estável.

A força de empuxo f é calculada a partir da força vetorial $\boldsymbol{F}$ calculada pelo controlador de orientação da carga e apresentada na seção 3.3.2. A força de empuxo que o quadricóptero é capaz de executar é obtida projetando $\boldsymbol{F}$ no eixo normal ao plano do quadricóptero $\boldsymbol{b}_3$. Sabendo que $\boldsymbol{b}_3$ está relacionado com $\boldsymbol{e}_3$ por $\boldsymbol{b}_3 = \boldsymbol{R}\boldsymbol{e}_3$, então

$$f = \boldsymbol{F} \cdot \boldsymbol{R}\boldsymbol{e}_3. \tag{3.20}$$

3.3.2 Controlador de orientação da carga

Este modo de controle é responsável por rastrear uma orientação desejada para a carga através do vetor unitário $\boldsymbol{p}_d \in S^2$. A saída do controlador é a matriz de rotação $\boldsymbol{R}_c$ que serve de entrada para o controlador de orientação do quadricóptero, substituindo $\boldsymbol{R}_d$. O mesmo ocorre para a matriz anti-simétrica correspondente à velocidade angular desejada que é substituída por

$$\hat{\boldsymbol{\Omega}}_c = \boldsymbol{R}_c^T \dot{\boldsymbol{R}}_c. \tag{3.21}$$

A matriz de rotação $\boldsymbol{R}_c$ é construída a partir de três vetores ortonormais $\boldsymbol{b}_{1_c}$, $\boldsymbol{b}_{2_c}$ e $\boldsymbol{b}_{3_c}$ conforme

$$\boldsymbol{R}_c = [\boldsymbol{b}_{1_c}, \boldsymbol{b}_{2_c}, \boldsymbol{b}_{3_c}]. \tag{3.22}$$

Estes vetores representam a orientação em que se espera que o sistema de coordenadas do corpo do quadricóptero atinja. O processo de obtenção dos vetores que compõem $\boldsymbol{R}_c$ está ilustrado na Figura 4 e é descrito a seguir.

O vetor $\boldsymbol{b}_{3_c}$ aponta na direção da força F desejada que o quadricóptero execute, ou seja,

$$\boldsymbol{b}_{3_c} = \frac{\boldsymbol{F}}{||\boldsymbol{F}||}. \tag{3.23}$$

Este vetor por si só não define um sistema de coordenadas, para isso, é necessário um vetor extra $\boldsymbol{b}_{1_d}$ que é usado para definir a direção desejada do eixo $\boldsymbol{b}_1$ do quadricóptero. O sistema de geração de trajetória é responsável por fornecer $\boldsymbol{b}_{1_d}$ com a restrição de que

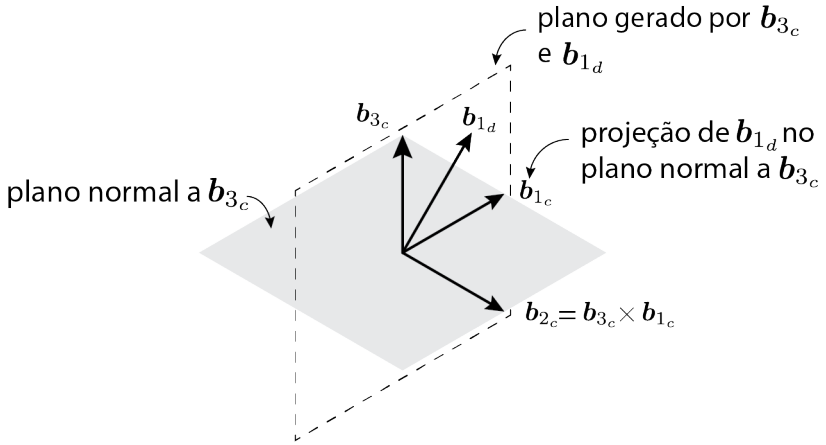


Figura 4 – Geração do sistema de coordenadas que constituem a matriz de rotação $\boldsymbol{R}_c$. O vetor unitário $\boldsymbol{b}_{3_c}$ é calculado em (3.23), $\boldsymbol{b}_{1_d}$ é escolhido arbitrariamente não sendo paralelo a $\boldsymbol{b}_{3_c}$ e está relacionado com a guinada, que define a direção que quadricóptero estará "olhando". Com estes dois vetores, é possível obter $\boldsymbol{b}_{1_c}$ e $\boldsymbol{b}_{2_c}$.

não seja paralelo a $\boldsymbol{b}_{3_c}$. Então, o vetor $\boldsymbol{b}_{1_c}$ é obtido com a projeção de $\boldsymbol{b}_{1_d}$ no plano gerado por $\boldsymbol{b}_{3_c}$,

$$\boldsymbol{b}_{1_c} = -\frac{1}{||\boldsymbol{b}_{3_c} \times \boldsymbol{b}_{1_d}||}(\boldsymbol{b}_{3_c} \times (\boldsymbol{b}_{3_c} \times \boldsymbol{b}_{1_d})). \tag{3.24}$$

Em seguida, basta encontrar $\boldsymbol{b}_{2_c}$ como produto vetorial dos dois vetores já obtidos, ou seja,

$$\boldsymbol{b}_{2_c} = \boldsymbol{b}_{3_c} \times \boldsymbol{b}_{1_c}. \tag{3.25}$$

Para o controlador de orientação da carga estar completo, resta definir $\boldsymbol{F}$. Esta força é obtida com a combinação de uma força de realimentação $\boldsymbol{F}_{pd}$, proporcional e derivativa, uma força de *feedforward* $\boldsymbol{F}_{ff}$, para garantir o rastreamento da trajetória, e uma força normal $\boldsymbol{F}_n$, para garantir que a orientação do quadricóptero esteja bem definida. O processo de obtenção destas forças para sistemas mecânicos genéricos é apresentado em Bullo e Lewis (2004) na seção 11.2.5 e definidas em Sreenath, Lee e Kumar (2013) para o caso do sistema quadricóptero-carga. As forças são definidas da seguinte forma

$$\boldsymbol{F}_n = -(\boldsymbol{p}_d \cdot \boldsymbol{p})\boldsymbol{p}, \tag{3.26}$$

$$\boldsymbol{F}_{pd} = -k_{\boldsymbol{p}}\boldsymbol{e}_{\boldsymbol{p}} - k_{\boldsymbol{\omega}}\boldsymbol{e}_{\dot{\boldsymbol{p}}}, \tag{3.27}$$

$$\boldsymbol{F}_{ff} = m_Q l(\boldsymbol{p} \cdot (\boldsymbol{p}_d \times \dot{\boldsymbol{p}}_d))(\boldsymbol{p} \times \dot{\boldsymbol{p}}) + m_Q l(\boldsymbol{p}_d \times \ddot{\boldsymbol{p}}_d) \times \boldsymbol{p}, \tag{3.28}$$

em que $\boldsymbol{p}_d$ corresponde à entrada do controlador e representa o valor de orientação da carga desejado.

Então,

$$\boldsymbol{F} = \boldsymbol{F}_n - \boldsymbol{F}_{pd} - \boldsymbol{F}_{ff}. \tag{3.29}$$

Sreenath, Lee e Kumar (2013) provam que o ponto de equilíbrio $(\boldsymbol{e_p}, \boldsymbol{e_{\dot{p}}}, \boldsymbol{e_R}, \boldsymbol{e_\Omega}) = (\boldsymbol{0}, \boldsymbol{0}, \boldsymbol{0}, \boldsymbol{0})$ é exponencialmente estável desde que as condições iniciais satisfaçam

$$\Psi_{\boldsymbol{p}}(\boldsymbol{p}(0), \boldsymbol{p}_d(0)) < 2, \tag{3.30}$$

$$||\boldsymbol{e_{\dot{p}}}(0)||^2 < \frac{2}{m_Q l} k_{\boldsymbol{p}}(2 - \Psi_{\boldsymbol{p}}(\boldsymbol{p}(0), \boldsymbol{p}_d(0))), \tag{3.31}$$

e que também satisfaçam as condições de estabilidade do controlador de orientação do quadricóptero (3.18) e (3.19). As regiões de atração para a orientação da carga e para a orientação do quadricóptero são bastante semelhantes. O erro de orientação da carga deve ser menor que 180°, independente do sentido de rotação. Isto resulta em estabilidade exponencial quase global.

3.3.3 Controlador de posição da carga

Este modo de controle é responsável por rastrear a posição desejada da carga $\boldsymbol{x}_L^d$. A saída do controlador é o vetor de orientação da carga $\boldsymbol{p}_c$ que serve de entrada para o controlador de orientação da carga substituindo $\boldsymbol{p}_d$.

Partindo da definição de erro apresentada em (C.2) e (C.4) segue que

$$\dot{\boldsymbol{e}}_v = \ddot{\boldsymbol{x}}_L - \ddot{\boldsymbol{x}}_L^d = \dot{\boldsymbol{v}}_L - \ddot{\boldsymbol{x}}_L^d. \tag{3.32}$$

Substituindo (3.32) na equação de movimento referente ao movimento da carga (3.8), tem-se

$$\begin{aligned}(m_L + m_Q)(\dot{\boldsymbol{e}}_v + \ddot{\boldsymbol{x}}_L^d + g\boldsymbol{e}_3) &= (\boldsymbol{p} \cdot f\boldsymbol{R}\boldsymbol{e}_3 - m_Q l(\dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}}))\boldsymbol{p} \\ (m_L + m_Q)\dot{\boldsymbol{e}}_v &= -(m_L + m_Q)(\ddot{\boldsymbol{x}}_L^d + g\boldsymbol{e}_3) + (\boldsymbol{p} \cdot f\boldsymbol{R}\boldsymbol{e}_3 - m_Q l(\dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}}))\boldsymbol{p}.\end{aligned} \tag{3.33}$$

Partindo desta dinâmica de erro, Sreenath, Lee e Kumar (2013) definem a orientação da carga calculada como

$$\boldsymbol{p}_c = -\frac{\boldsymbol{A}}{||\boldsymbol{A}||}, \tag{3.34}$$

em que

$$\boldsymbol{A} = -k_x\boldsymbol{e}_x - k_v\boldsymbol{e}_v + (m_Q + m_L)(\ddot{\boldsymbol{x}}_L^d + g\boldsymbol{e}_3) + m_Q l(\dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}})\boldsymbol{p}, \tag{3.35}$$

em que k_x e k_v são constantes positivas e assume-se que $||\boldsymbol{A}|| \neq 0$. Além disso, substitui-se (3.26) pela seguinte definição

$$\boldsymbol{F}_n = (\boldsymbol{A} \cdot \boldsymbol{p})\boldsymbol{p} \tag{3.36}$$

É possível ver que $\boldsymbol{A}$ é definido partindo do cancelamento de termos em (3.32) com um termo proporcional e um derivativo. Esta comparação serve apenas para ilustrar de onde $\boldsymbol{p}_c$ é obtido, porém a prova de estabilidade é muito mais complexa e é apresentada em Sreenath, Lee e Kumar (2013). Neste artigo também é demonstrado que o ponto de

equilíbrio $(\boldsymbol{e}_x, \boldsymbol{e}_v, \boldsymbol{e}_{\boldsymbol{p}}, \boldsymbol{e}_{\dot{\boldsymbol{p}}}, \boldsymbol{e}_{\boldsymbol{R}}, \boldsymbol{e}_{\boldsymbol{\Omega}}) = (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0})$ é exponencialmente estável dado que as condições iniciais satisfaçam

$$\Psi_p(\boldsymbol{p}(0), \boldsymbol{p}_c(0)) < \psi_1 < 1 \tag{3.37}$$

$$||\boldsymbol{e}_x(0)|| < e_{x_{max}}, \tag{3.38}$$

em que ψ_1 e $e_{x_{max}}$ são constantes que fazem parte da prova de estabilidade em Sreenath, Lee e Kumar (2013).

Assim, a região de atração do controlador é caracterizada por

$$||\boldsymbol{e}_{\dot{\boldsymbol{p}}}(0)||^2 < \frac{2}{m_Q l} k_{\boldsymbol{p}}(\psi_1 - \Psi_{\boldsymbol{p}}(\boldsymbol{p}(0), \boldsymbol{p}_c(0)), \tag{3.39}$$

bem como (3.38) e a regiões de atração dos controlador de orientação do quadricóptero (3.18), (3.19).

3.4 Estimação das derivadas

Uma característica de um sistema de controle recursivo, como o implementado para o controlador geométrico, é que cada subsistema necessita das derivadas das atuações intermediárias calculadas. Para o sistema em questão, as atuações são $\boldsymbol{R}_c$ e $\boldsymbol{p}_c$ e são utilizadas as primeiras e segundas derivadas destes termos.

Estas derivadas podem ser obtidas de forma numérica pela própria definição de derivada. Porém, não é uma boa opção na presença de ruído ou até mesmo na partida do sistema quando ocorrem derivadas altas que podem desestabilizar o sistema.

Outra opção é calcular a derivada analiticamente. A derivada analítica não é afetada pela presença de ruídos, porém necessita do conhecimento prévio da trajetória e também requer um trabalho manual para o cálculo das derivadas de funções bastante complexas. Outro problema desta técnica é que por utilizar a derivada analítica da trajetória, o controlador tem mais dificuldade em ajustar erros já que a derivada utilizada não é a que está sendo executada pelo sistema. Uma variante desta abordagem é utilizada por Sreenath, Lee e Kumar (2013) com a justificativa que deixa o sistema de controle muito mais simples.

Uma boa opção é implementar um estimador para as variáveis de comando e suas derivadas. Neste caso, a solução pode ser mais robusta com relação a possíveis ruídos e também com relação à partida do sistema, além de ser independente da trajetória.

Um filtro comumente utilizado para calcular as derivadas que surgem em um sistema de controle baseado em recursão é chamado de filtro de comando (FARRELL et al., 2009). Uma abordagem simples de um filtro de comando é assumir que as variáveis são de segunda ordem e implementar um filtro com as seguintes equações no espaço de

estados

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -2\zeta\omega_n - \omega_n^2(x_1 - x_c), \end{aligned} \tag{3.40}$$

em que ζ corresponde ao fator de amortecimento, ω_n à frequência natural do filtro e x_c à variável a ser estimada. Desta forma, x_1 corresponde ao valor estimado de x_c e x_2 corresponde ao valor estimado de $\dot{x}_c$. A segunda derivada de x_c é obtida direto de $\dot{x}_2$ em (3.40).

As vantagens em estimar a variável calculada com (3.40) são que não há dependência da trajetória, ruído de alta frequência afeta pouco a estimação e também não há problemas na partida do sistema já que o filtro irá suavizar as ações de controle que normalmente causariam derivadas altas. A desvantagem é que há dois parâmetros para ajustar que podem influenciar positivamente ou negativamente a estabilidade do sistema de controle.

3.5 Árvore Aleatória de Exploração Rápida (RRT)

O algoritmo RRT consiste em uma estrutura de dados eficiente e um esquema de amostragem capaz de rapidamente buscar em espaços de alta dimensão que possuem restrições algébricas, que surgem de obstáculos, e restrições diferenciais, que podem ser não-holonômicas e dinâmicas (LAVALLE; JR, 2000). A ideia principal do algoritmo é enviesar o crescimento da árvore para regiões inexploradas do espaço ao amostrar pontos aleatoriamente no espaço de busca e fazer a árvore crescer na direção das configurações sorteadas. Além disso, a RRT é um algoritmo probabilisticamente completo, em outras palavras, a probabilidade de encontrar uma solução tende a 1 quando o número de iterações tende a infinito (LAVALLE et al., 2001).

O crescimento da RRT está ilustrado na Figura 5. Nesta figura, a árvore já contém nós expandidos e os passos executados em uma iteração do algoritmo são explicados a seguir. Primeiramente, um ponto no espaço de busca é selecionado de forma aleatória e denominado $\boldsymbol{q}_{rand}$. O espaço de busca pode ser o de configuração, de trabalho, ou outro desejado. Em seguida, o nó da árvore mais próximo é selecionado e denominado $\boldsymbol{q}_{near}$. De acordo com Shkolnik e Tedrake (2009), a métrica usual nos algoritmos baseados na RRT para medir a distância costuma ser a Euclidiana, mas há trabalhos que utilizam outras técnicas, como a função de custo do LQR utilizada por Perez et al. (2012).

O próximo passo é fazer a árvore crescer em direção a $\boldsymbol{q}_{rand}$ a partir de $\boldsymbol{q}_{near}$. Para sistemas com restrições dinâmicas, o algoritmo clássico envolve selecionar uma entrada (u) dentro das restrições de atuação que leve o sistema em direção a $\boldsymbol{q}_{rand}$. Então, o próximo estado $\boldsymbol{q}_{new}$ é obtido integrando o modelo do sistema partindo do estado $\boldsymbol{q}_{near}$ durante um intervalo de tempo Δt.

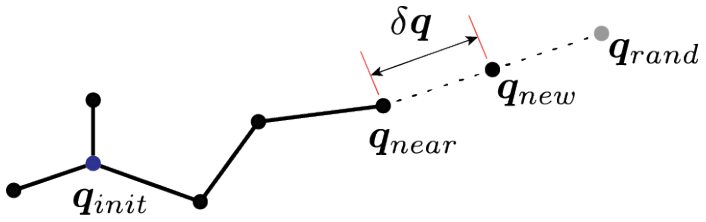


Figura 5 – Crescimento da RRT após 5 expansões de nós.

Existem diversas formas de se obter a entrada que faça o sistema evoluir em direção a $\boldsymbol{q}_{rand}$ para obter $\boldsymbol{q}_{new}$. Uma abordagem utilizada por Shkolnik, Walter e Tedrake (2009) consiste em discretizar o espaço de ações, obter o conjunto de estados alcançáveis partindo de $\boldsymbol{q}_{near}$ e então verificar se há algum estado que evolua o sistema em direção a $\boldsymbol{q}_{rand}$. Outra forma é utilizar um controlador específico para o sistema, como realizado por Karaman et al. (2011) ou um controlador mais genérico como o LQR aplicado por Perez et al. (2012).

Uma vez que $\boldsymbol{q}_{new}$ é obtido, basta checar se há colisões entre os estados $\boldsymbol{q}_{near}$ e $\boldsymbol{q}_{new}$. Uma forma de simplificar a checagem é fazer o intervalo de integração (Δt) do algoritmo pequeno o suficiente para que a variação entre dois estados ($\delta \boldsymbol{q}$) seja pequena de forma que seja possível checar se apenas $\boldsymbol{q}_{new}$ colide com algum obstáculo no mapa.

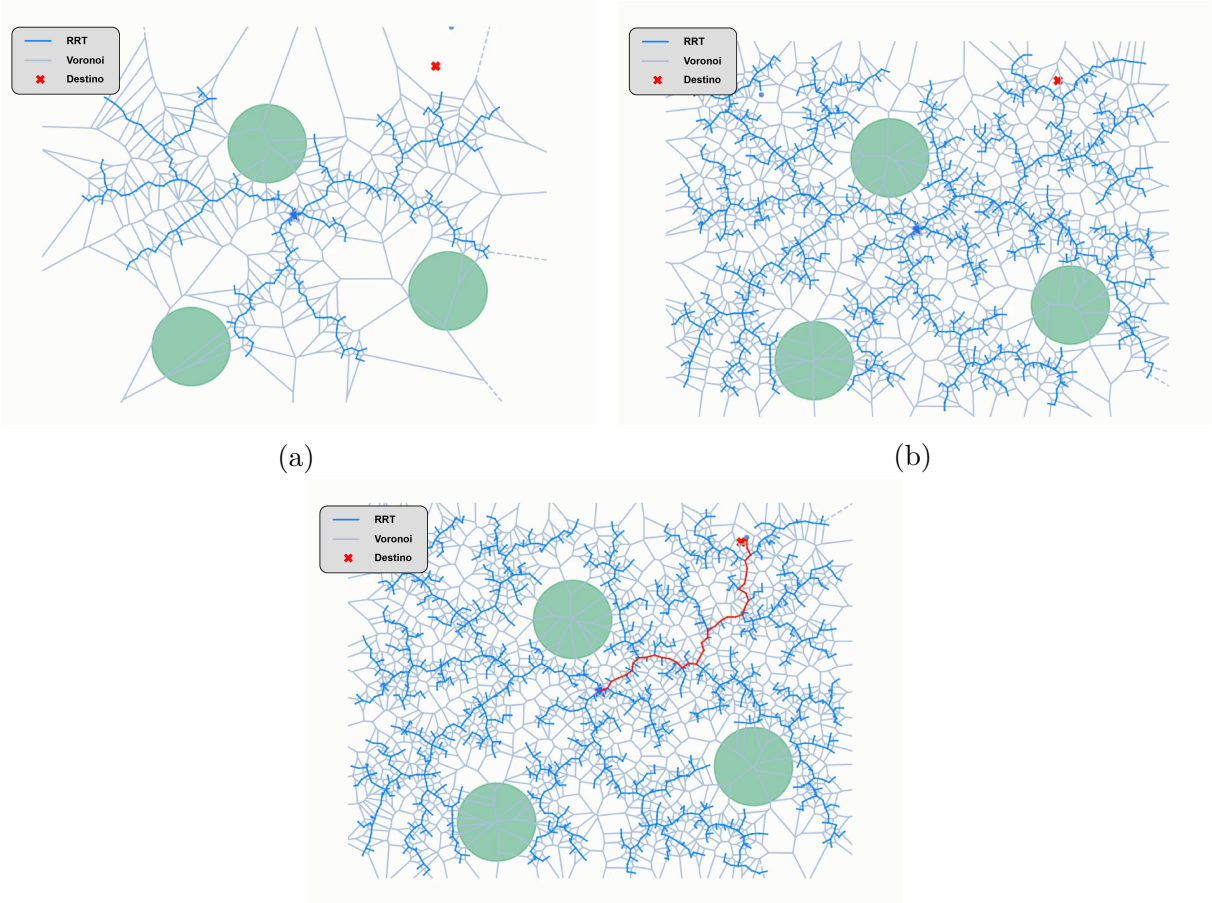
Se não houver colisão, então $\boldsymbol{q}_{new}$ é adicionado à arvore e uma nova iteração do algoritmo ocorre. Todos os passos mencionados são mostrados no Algoritmo 1. Como entrada do algoritmo, tem-se a configuração inicial (q_{init}), o número máximo de iterações durante a busca (K) e o intervalo de integração (Δt). Ao término do algoritmo, o resultado é uma árvore que contém trajetórias sem colisões para diversos locais do ambiente.

Algoritmo 1: $\mathcal{T} \leftarrow RRTClássica(q_{init}, N, \Delta t)$

```
T ← InicializarÁrvore(q_init);
para j = 1 até N faça
    q_rand ← EstadoAleatório();
    q_near ← VizinhoMaisPróximo(q_rand, T);
    u ← SelecionarEntrada(q_rand, q_near);
    q_new ← NovoEstado(q_near, u, Δt);
    se LivreDeObstáculo(q_near, q_new) então
        T ← InserirNó(q_near, q_new, u, T);
    fim
fim
```

Na Figura 6, é ilustrado o crescimento em três momentos de uma RRT clássica em um ambiente com obstáculos. Além disso, é ilustrado o Diagrama de Voronoi gerado pelos nós da RRT para facilitar o entendimento do porquê a RRT é tão eficiente em explorar o espaço (KUFFNER; LAVALLE, 2000). O Diagrama de Voronoi decompõe o espaço em células a partir de um conjunto de sementes geradoras de forma que qualquer ponto amostrado em uma célula está mais próximo de seu gerador que qualquer outro gerador

em outras células. No caso da RRT, os geradores do Diagrama de Voronoi são os seus nós. Desta forma, uma célula neste diagrama corresponde a uma região em que qualquer ponto amostrado dentro dela terá o seu nó gerador como vizinho mais próximo.



(a) (b)

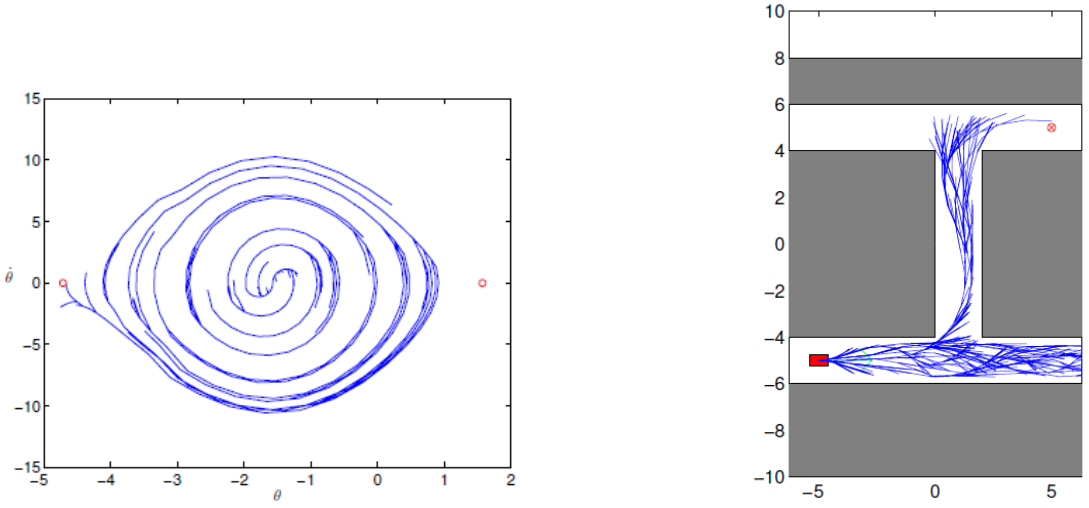
(c)

Figura 6 – Crescimento da RRT clássica em três momentos: (a) a RRT começa a se expandir em direção às regiões de Voronoi maiores; (b) a RRT continua se expandindo com ramos menores; (c) a RRT amostra um ponto próximo do destino e retorna uma trajetória. Os três círculos representam obstáculos.

Com esse entendimento, os nós que estão nas regiões de Voronoi com maior área são mais prováveis de serem selecionados para a expansão da árvore. Isto ocorre porque qualquer ponto amostrado nesta região terá como seu vizinho mais próximo o nó dentro dela. No começo da expansão da árvore, as regiões de Voronoi possuem áreas maiores e a RRT rapidamente se expande cobrindo as regiões inexploradas, como se vê na Figura 6a.

Com o crescimento da RRT, o número de nós aumenta e, consequentemente, as regiões de Voronoi aumentam em quantidade e reduzem em área. Neste momento, as áreas das regiões de Voronoi tendem a ficar uniformes e o algoritmo começa a criar ramos que preenchem os espaços vazios, como está ilustrado na Figura 6b. O algoritmo é finalizado assim que um nó é amostrado próximo o suficiente ao destino desejado e, então, a trajetória é retornada. Este momento está representado na Figura 6c.

Na Figura 7, são apresentados dois exemplos diferentes de aplicação de RRT feitos por Shkolnik, Walter e Tedrake (2009). Na Figura 7a, o sistema consiste em um pêndulo invertido com limite de torque. Na Figura 7b, o modelo consiste em um carro com quatro dimensões no espaço de estados. Ambas as buscas foram feitas no espaço de estados, mas as trajetórias do carro são mostradas no espaço de trabalho.



(a) (b)

Figura 7 – Exemplos de RRT: (a) de um pêndulo com trajetórias apresentadas no no espaço de estados e (b) de um carro com trajetórias apresentadas no espaço de trabalho. Ambos implementados por Shkolnik, Walter e Tedrake (2009).

Como mencionado previamente, a RRT tende a crescer em direção a regiões inexploradas do ambiente. Se for desejado que o algoritmo solucione uma tarefa específica, ou seja, encontre uma trajetória entre uma configuração inicial e uma desejada, é possível alterar o algoritmo de forma a também polarizar o crescimento da árvore em direção ao destino. Uma abordagem sugerida por LaValle e Jr (2001) consiste em explorar o espaço de busca crescendo duas árvores, uma partindo do início e outra partindo do destino. Metade do tempo o algoritmo cresce a árvore focando na exploração e a outra metade focando conectar as duas árvores. Esta abordagem tem como desvantagem a existência de de descontinuidade na trajetória no local em que as árvores conectam.

Outra abordagem menos eficiente, mas que não tem o problema de continuidade, consiste em amostrar com alguma probabilidade pequena p a configuração final desejada e com probabilidade $1 - p$ amostrar uma configuração aleatória. Esta modificação também é discutida por LaValle e Jr (2001) e aplicada em Shkolnik e Tedrake (2009). Além disso, pode-se adicionar um critério de parada quando a solução está próxima o suficiente da configuração desejada. Assim, o algoritmo retorna a solução assim que ela é encontrada. Ambas modificações são utilizadas por Shkolnik e Tedrake (2009).

Uma possível implementação da RRT considerando uma probabilidade de amostrar a configuração desejada p e uma etapa de verificação se uma solução foi encontrada está apresentada no Algoritmo 2. Os parâmetros adicionais são ρ, que representa uma distância mínima para detectar a solução encontrada, e P, que define a probabilidade de escolha do destino ao invés de um ponto aleatório.

Algoritmo 2: $\mathcal{T} \leftarrow RRTClássica(q_{init}, N, \Delta t, \rho, P)$

```
T ← InicializarÁrvore(q_init);
para j = 1 até N faça
    se com probabilidade P então
        q_rand ← q_goal;
    senão
        q_rand ← EstadoAleatório();
    fim
    q_near ← VizinhoMaisPróximo(q_rand, T);
    u ← SelecionarEntrada(q_rand, q_near);
    q_new ← NovoEstado(q_near, u, Δt);
    se LivreDeObstáculo(q_near, q_new) então
        T ← InserirNó(q_near, q_new, u, T);
        se distância(q_new, q_goal) < ρ então
            retorna T
        fim
    fim
fim
```

4 Desenvolvimento do Trabalho e Abordagem Proposta

Neste capítulo, é apresentado o desenvolvimento do trabalho que envolveu um estudo inicial do modelo simplificado planar do sistema quadricóptero-carga com o intuito de entender o comportamento do sistema para implementar o sistema tridimensional. Então, é apresentado o desenvolvimento e o desempenho do controlador escolhido. Por fim, é proposto um algoritmo baseado na RRT para o planejamento de movimento utilizando o sistema quadricóptero carga.

4.1 Estudo inicial do sistema quadricóptero-carga planar

Os primeiros passos deste trabalho foram focados em entender o comportamento do sistema quadricóptero-carga. Com este objetivo em mente, foi decidido levantar o modelo de um quadricóptero com carga limitado a um plano. Desta forma, a configuração seria apenas a posição bidimensional da carga, o ângulo do quadricóptero e o da carga, resultando em apenas 4 variáveis e 2 motores para atuar o sistema no plano yz. Este modelo está apresentado na Figura 8. Com um sistema simplificado, seria mais fácil implementar um sistema de controle, visualizar a evolução do sistema e entender os desafios que iriam surgir.

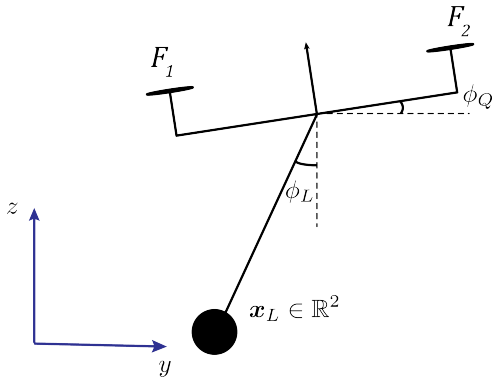


Figura 8 – Quadricóptero planar com carga suspensa.

O levantamento do modelo planar resultou em

$$\ddot{y}_L = \frac{1}{m_Q + m_L}\Big[\Big(- m_Q l\dot{\phi}_L^2 - Fcos(\phi_L - \phi_Q)\Big)sin(\phi_L)\Big], \tag{4.1}$$

$$\ddot{z}_L = \frac{1}{m_Q + m_L}\Big[\Big(m_Q l\dot{\phi}_L^2 + Fcos(\phi_L - \phi_Q)\Big)cos(\phi_L)\Big] - g, \tag{4.2}$$

$$\ddot{\phi}_L = \frac{1}{m_Q l}\Big(- Fsin(\phi_L - \phi_Q)\Big), \tag{4.3}$$

$$\ddot{\phi}_Q = \frac{M}{J_Q}, \tag{4.4}$$

em que F e M são as entradas de força e momento resultantes do sistema geradas pelos dois motores, m_Q e m_L são as massas do quadricóptero e da carga, g é a aceleração da gravidade, l é o comprimento do cabo, y e z a posição da carga e ϕ_Q e ϕ_L os ângulos do quadricóptero e da carga.

A primeira impressão foi que se o sistema simplificado já continha diversos termos trigonométricos, o modelo completo seria mais complexo. Ao continuar o estudo do modelo, o sistema foi linearizado e controlado utilizando um controlador linear proporcional e derivativo. Conforme esperado, o sistema foi estabilizado em torno do ponto de equilíbrio mas não seria possível realizar manobras complexas com um controlador simples. Então, o foco foi em implementar um controlador não linear para o modelo planar e estendê-lo para o modelo tridimensional.

Um controlador não linear para o sistema quadricóptero-carga planar é apresentado por Sreenath, Michael e Kumar (2013). Nesse artigo, é mostrado que o controlador é capaz de estabilizar a carga e portanto foi implementado para que fosse possível verificar sua robustez. Após a implementação, foi visto que apesar do controlador ser capaz de estabilizar o sistema, a equação de controle tinha dificuldade em estabilizar o sistema para configurações muito distantes do ponto de equilíbrio.

Após avaliar o comportamento definido pelas equações, foi visto que o problema estava na equação que define o ângulo calculado do quadricóptero a partir do ângulo da carga

$$\phi_Q^c = \phi_L + sin^{-1}\Big(- k_p^L e_l - k_d^L \dot{e}_l + \frac{\ddot{\phi}_L^d m_Q l}{f}\Big), \tag{4.5}$$

em que $e_L = \phi_L - \phi_L^d$ e k_p^L e k_d^L são constantes positivas.

Esta equação possui duas características que afetam a eficiência do controlador. A primeira é que a função $sin^{-1}(x)$ é definida apenas no intervalo $[-1, 1]$. Por este motivo, as acelerações desejadas ($\ddot{\phi}_L^d$) devem ser pequenas e as constantes do controlador devem ser limitadas para satisfazer o domínio da função. O segundo fato é que a força f é obtida a partir da projeção de um vetor que está relacionado com a direção de movimentação

desejada no vetor normal ao quadricóptero. Esta projeção pode ser próxima de zero a depender do estado do sistema e do estado desejado. Neste caso, f pode assumir valores próximos de zero causando problemas de singularidade em (4.5).

Estes testes foram importantes para estabelecer os critérios de modelo e controladores escolhidos para o desenvolvimento do trabalho. Considerando a complexidade das equações obtidas de (4.1) a (4.4) e a restrição de domínio das funções de controle causadas pelas funções trigonométricas, foi entendido que o modelo precisaria ser parametrizado de uma forma mais eficiente, como *quaternions*, ou precisaria ser obtido um modelo independente de parametrizações.

Então, durante a revisão foi encontrado o trabalho de Sreenath, Lee e Kumar (2013) em que propõem um modelo tridimensional do sistema quadricóptero-carga livre de parametrizações e também um controlador geométrico não linear robusto. Eles resolvem exatamente as questões que foram levantadas durante os testes com o sistema planar. As equações dinâmicas do sistema apresentadas por eles são livres de funções trigonométricas e definidas de forma vetorial e compacta, simplificando as equações. Isto facilitou o desenvolvimento das equações de controle mais robustas. O modelo é apresentado na Seção 3.2 e o projeto dos controladores na Seção 3.3.

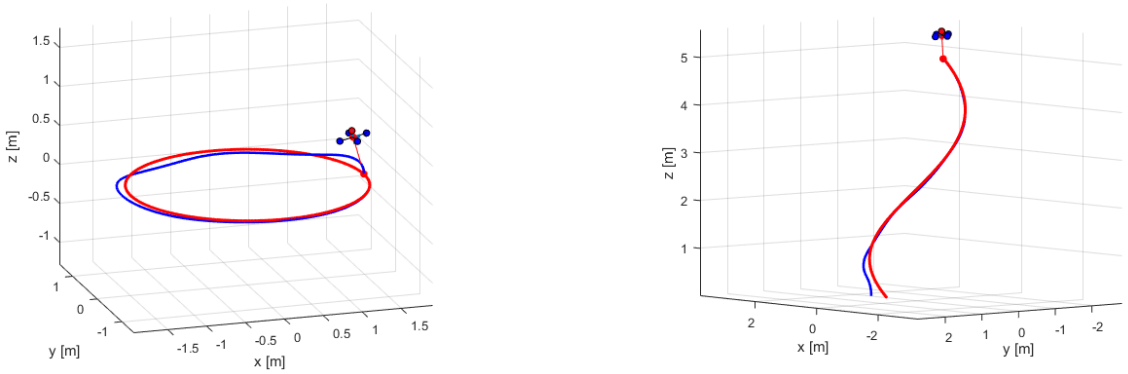
A partir do entendimento do modelo e do comportamento do sistema planar, foi iniciada a modelagem tridimensional do sistema quadricóptero-carga e a implementação do controlador não linear proposto por Sreenath, Lee e Kumar (2013).

4.2 Implementação do Sistema Quadricóptero-Carga Controlado

Com o objetivo de se obter um comportamento do modelo mais próximo da realidade, os parâmetros utilizados para a implementação do sistema quadricóptero-carga deviam ser escolhidos baseados em um quadricóptero real. Considerando a aplicação de transporte de cargas suspensas, foi decidido utilizar os parâmetros do quadricóptero Hummingbird cujos parâmetros estão disponíveis por ser bastante utilizado na literatura. Alguns parâmetros são divulgados por Michael et al. (2010), onde apresentam uma plataforma de testes de quadricópteros de pequeno porte. Outros são divulgados em Powers, Mellinger e Kumar (2015) e os remanescentes foram obtidos no *datasheet* do quadricóptero disponibilizado por ASCTEC (2015). O compilado de parâmetros do sistema implementado estão apresentados na Tabela 3.

Após a definição dos parâmetros e implementação das equações do modelo, (3.7) a (3.12), e do controlador, o sistema foi simulado com diversas trajetórias e ganhos para o controlador e para as constantes de estimação das derivadas. Entre as trajetórias testadas, estão uma circular e uma helicoidal ilustradas na Figura 9. É possível ver um

erro transitório que ocorre devido ao sistema estar em repouso ou com erro de posição inicial, mas após alguns segundos o sistema é capaz de acompanhar a trajetória desejada. Nas duas trajetórias ilustradas, a simulação ocorreu por 5 s.



(a) (b)

Figura 9 – Trajetórias executadas pelo sistema quadricóptero carga com o controlador geométrico: (a) trajetória circular de raio 1.5 m e período 5 s e (b) trajetória helicoidal de raio 1 m, período 2π s e velocidade ascendente de 1 m/s.

Foram diversos testes até encontrar ganhos capazes de fazer o sistema rastrear as trajetórias testadas com *overshoot* moderado e respeitando os limites de atuação e de velocidades fornecidos pelo fabricante do quadricóptero. As constantes dos controladores

Tabela 3 – Parâmetros do quadricóptero com carga suspensa.

Parâmetro	**Símbolo**	**Valor**
Altura	h	0,0855 m
Comprimento do braço	L	0,175 m
Comprimento do cabo	l	0,5 m
Massa do quadricóptero	m_Q	0,835 kg
Massa da carga	m_L	0,088 kg
Momento de inércia em X	I_{xx}	$2{,}32 \cdot 10^{-3}\,\mathrm{kg\,m^{-2}}$
Momento de inércia em Y	I_{yy}	$2{,}32 \cdot 10^{-3}\,\mathrm{kg\,m^{-2}}$
Momento de inércia em Z	I_{zz}	$4{,}00 \cdot 10^{-3}\,\mathrm{kg\,m^{-2}}$
Velocidade máxima do motor	w_{max}	7800 rpm
Velocidade mínima do motor	w_{min}	1200 rpm
Constante de força do motor	k_F	$6{,}11 \cdot 10^{-8}\,\mathrm{N\,rpm^{-2}}$
Constante de momento do motor	k_M	$1{,}5 \cdot 10^{-9}\,\mathrm{N\,m\,rpm^{-2}}$

Tabela 4 – Ganhos do controlador geométrico.

Constante	**Valor**
k_R	$diag(1{,}40; 1{,}40; 0{,}40)$
k_Ω	$diag(0{,}12; 0{,}12; 0{,}08)$
k_q	9
k_ω	7,5
k_x	$diag(5{,}40; 5{,}40; 5{,}85)$
k_v	$diag(6{,}00; 6{,}00; 4{,}5)$

$diag(.;.;.)$ especifica os elementos de uma matriz diagonal.

estão dispostas na Tabela 4. Com relação ao estimador das derivadas, dado por (3.40), o parâmetro ζ foi 0.98 para a carga e o quadricóptero e ω_n foi 7 e 75 para a carga e para o quadricóptero, respectivamente.

É importante salientar que apesar das constantes apresentadas nas provas de estabilidade serem do tipo escalar, os controladores são capazes de estabilizar o sistema com constantes matriciais como pode ser visto em Tang, Wüest e Kumar (2018). Como o sistema possui dinâmicas diferentes em cada dimensão, é mais vantajoso utilizar matrizes diagonais para dar ganhos adequados. Por exemplo, o sistema possui dinâmica bem diferente no eixo em que a gravidade atua com relação aos outros dois eixos do sistema de coordenadas inercial e por isso necessita de valores de ganhos diferentes.

Para ilustrar as capacidades e limitações do sistema com os ganhos escolhidos, são apresentadas as execuções de 3 trajetórias circulares na Figura 10. Em todas as imagens o quadricóptero executou 3 voltas, porém com períodos diferentes que requerem diferentes esforços do controlador para rastrear a trajetória.

Na Figura 10a, o período é de 5 s e o controlador é capaz de rastrear a trajetória sem dificuldades e com pouca sobrepassagem. Na Figura 10b, o período é de 4 s e o controlador leva mais tempo para alcançar a trajetória desejada com um pouco mais de sobrepassagem que o primeiro caso. Na Figura 10c, o período é de 3 s e o controlador tem dificuldades em rastrear a trajetória e o sistema sofre com sobrepassagem de aproximadamente 1,5 m e após as três voltas, a carga ainda não atinge a trajetória ficando um erro estacionário de aproximadamente 20 cm no plano xy. Além disso, devido à perda de sustentação do quadricóptero por causa da inclinação necessária para executar a trajetória, há um erro de altitude de aproximadamente 10 cm.

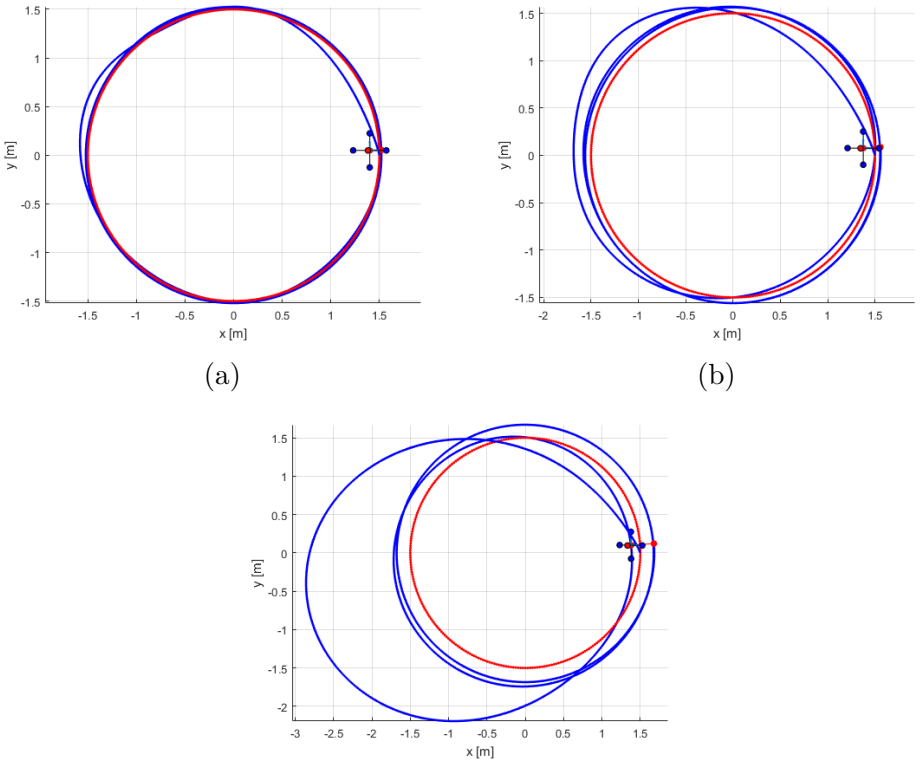


Figura 10 – Trajetórias circulares com períodos (T) diferentes: (a) $T = 5\ s$ (b) $T = 4\ s$ e (c) $T = 3\ s$. Todas as trajetórias foram executadas por $3T\ s$.

Ainda foram realizados mais testes inspirados por Sreenath, Lee e Kumar (2013) em que foi possível verificar que o controlador é capaz de estabilizar o sistema em situações extremas como o quadricóptero e a carga rotacionados 178°.

Todos os testes realizados serviram para entender as capacidades e limitações do modelo e do controlador e foi de fato verificada a robustez do controlador geométrico em diversas situações. Então foi decidido utilizar o controlador em situações em que o quadricóptero tivesse que realizar movimentos bastante agressivos para executar uma tarefa, como o problema da janela curta que é bastante utilizado na literatura. Para tal, seria necessário implementar algum planejador de movimento proposto na literatura ou propor uma abordagem diferente.

Considerando as vantagens e desvantagens das técnicas apresentadas na revisão bibliográfica acerca do planejamento de movimento para resolver a tarefa da janela curta, foi decidido então propor um planejador de movimento baseado no controlador geométrico.

4.3 Planejamento de Trajetórias Baseado no Controlador Geométrico

Diversas abordagens de controle e planejamento de movimento são apresentadas no Capítulo 2 referente à revisão bibliográfica. Entre as propostas apresentadas, destacam-se três trabalhos. O de Crousaz, Farshidian e Buchli (2014) apresenta um algoritmo de aprendizado iterativo para o problema da janela curta. O de Tang e Kumar (2015) utiliza um gerador de trajetória baseado em programação quadrática e o mesmo controlador utilizado nessa dissertação. Por fim, o trabalho de Geisert e Mansard (2016), por também resolver o problema utilizando programação quadrática e sugerir uma abordagem com controle preditivo para a solução *online* da trajetória.

Todas as técnicas citadas resolvem o problema tentando assegurar que as equações a serem otimizadas sejam o mais próximo do convexo possível para que as minimizações das funções de custo possam convergir sem dificuldades. Para tal, os obstáculos são bastante simplificados para satisfazer às restrições dos algoritmos utilizados.

Neste contexto, foi decidido então desenvolver um planejador de movimento que não tivesse os problemas encontrados na literatura. Para tal, seria necessário implementar um sistema de planejamento que não dependesse da otimização de uma função de custo e que não fosse necessário limitar o formato dos obstáculos. Além disso, a técnica deveria ser capaz de resolver problemas de alta dimensão e a solução precisaria respeitar as restrições cinemáticas e dinâmicas para que fosse possível encontrar uma trajetória factível.

Ao listar tais características, foi verificado que algoritmos baseados em amostragem aleatória satisfazem as necessidades do problema, já que comumente não necessitam de etapas de minimização de funções de custo. Além disso, ao considerar a necessidade de respeitar as restrições cinemáticas e dinâmica do sistema com um espaço de busca de alta dimensão, foi visto que a técnica que mais se adéqua ao problema é a RRT. Esta é uma técnica probabilística que utiliza amostragens aleatórias para resolver problemas de alta dimensão e permite facilmente a adição de restrições dinâmicas. Contudo, as soluções obtidas são sub-ótimas e existe a possibilidade do algoritmo não encontrar uma solução em um determinado tempo mesmo que ela exista.

Com a técnica de planejamento mais adequada para o problema determinada, foi verificado na literatura se algo semelhante já havia sido proposto. Até o momento, não foram encontradas abordagens de planejamento utilizando RRT para o sistema quadricóptero-carga e foram encontradas poucas propostas de utilização de RRT para o planejamento de trajetórias de quadricóptero, sendo que as abordagens encontradas baseiam-se na linearização do sistema. Com o sistema linearizado, Perez et al. (2012)

utilizam um controlador LQR (*Linear Quadratic Regulator*) não só para atuar no sistema mas também para definir uma matriz de peso ótima para o cálculo do vizinho mais próximo.

Apesar da linearização permitir a utilização de um controlador que fornece uma matriz de custo ótima, para o problema analisado nesse trabalho, foi decidido que linearizar o sistema poderia limitar as aplicações do planejador de movimento já que o objetivo é permitir movimentações distantes do ponto de equilíbrio.

Sob tais condições, partiu-se para a implementação de uma RRT para a geração de trajetórias factíveis para o modelo dinâmico apresentados nas equações de (3.7) a (3.12). A implementação da RRT começou do algoritmo clássico apresentado por LaValle (1998) e descrito no Algoritmo 2. Então, o algoritmo foi modificado para melhor satisfazer às necessidades do problema investigado. Tais adaptações estão descritas nas Seções 4.4 a 4.8.

4.4 Definição do espaço de busca da RRT

O espaço de busca define a região em que a árvore será expandida. Quando se deseja considerar a dinâmica do sistema, o espaço de busca consiste no seu espaço de estados. Então, o primeiro desafio foi definir uma forma eficiente de sortear uma amostra aleatória no espaço de busca. Para sistemas simples, esta é uma etapa trivial já que basta amostrar de modo uniforme o espaço de estados do sistema. No caso de sistemas com altas dimensões e que possuem estados indesejáveis, como o de um quadricóptero transportando uma carga suspensa, amostrar uniformemente o espaço de estados seria uma forma ineficiente de busca.

O estado do quadricóptero, dado por $\boldsymbol{\mathcal{X}} = [\boldsymbol{x}_L, \boldsymbol{v}_L, \boldsymbol{p}, \boldsymbol{\omega}, \boldsymbol{R}, \boldsymbol{\Omega}]$, possui 24 elementos que podem ser parametrizados e amostrados em um espaço de 17 dimensões. Uma busca em um espaço desta magnitude pode afetar bastante o tempo de solução do algoritmo. Ainda assim, muitos dos estados selecionados podem representar configurações indesejadas, como a carga ou quadricóptero invertidos, que podem acabar sendo adicionados na árvore e a solução fazer com que a carga fosse arremessada em várias direções desnecessariamente.

Uma possível solução para tal problema seria definir intervalos de amostragem para cada elemento do estado do sistema de forma que os estados possíveis não incluíssem configurações com a carga ou quadricóptero invertidos, por exemplo. No entanto, as definições dos intervalos de alguns estados acabariam sendo feitas de forma arbitrária e aumentaria significativamente a quantidade de parâmetros do algoritmo. Com isto em mente, foram procuradas alternativas e a solução partiu do próprio controlador geométrico implementado.

Considerando que os ganhos do controlador tenham sido bem selecionados e que o modelo dinâmico simulado contenha as devidas restrições dinâmicas, basta selecionar um *setpoint* do controlador de maneira aleatória que, durante a simulação do sistema controlado, as equações de controle irão definir os estados intermediários necessários para atingir o *setpoint* amostrado. O *setpoint* do controlador é definido pela posição, velocidade e aceleração desejadas da carga e a guinada do quadricóptero. Entretanto, a guinada pode ser removida do espaço de busca, já que ela pode ser mantida constante sem afetar a capacidade de execução de tarefas do quadricóptero, e a aceleração também pode ser desconsiderada, pois é um termo de *feedforward* e apenas os termos de realimentação são necessários para guiar o sistema em direção à posição desejada.

Desta forma, quando for o *setpoint* for amostrado neste espaço de busca, será considerado que o vetor resultante é dado por

$$\mathbf{x} = [\boldsymbol{x}_L, \boldsymbol{v}_L]^T. \tag{4.6}$$

Sendo assim, o espaço de busca é reduzido para 6 dimensões (posição e velocidade da carga) cujos limites são fáceis de se determinar. A posição é limitada pelo espaço de trabalho desejado e as velocidades são limitadas de acordo com as máximas e mínimas do sistema em questão.

Com o espaço de busca definido, foi implementada a polarização da árvore em direção à posição final desejada.

4.5 Polarização da RRT em direção ao destino

Inicialmente, a RRT foi implementada com a polarização de probabilidade constante semelhante ao apresentado no Algoritmo 2. No entanto, foi decidido utilizar um método mais eficiente, pois a probabilidade constante define, já no início do algoritmo, se a RRT será focada em explorar o ambiente (atribuindo uma probabilidade pequena de seleção do destino) ou será focada em crescer a árvore em direção ao destino (atribuindo uma probabilidade maior de seleção do destino). Esta política de busca é bastante comum na área de aprendizado por reforço e possui o nome de ϵ-*greedy*. No primeiro caso, a RRT pode levar bastante tempo para solucionar o problema. No segundo caso, a RRT pode ficar presa em um mínimo local devido à baixa probabilidade de exploração do ambiente.

Uma forma mais eficiente é variar as probabilidades de seleção de uma amostra aleatória e do destino para permitir que a árvore explore o ambiente no início da busca e então comece a crescer em direção da solução final após a etapa de exploração. Desta forma, é possível acelerar a solução do algoritmo e reduzir a probabilidade do algoritmo convergir para um mínimo local. Esta política de busca é também considerada ϵ-*greedy*, porém com probabilidade variável.

A probabilidade de amostrar um valor aleatório no espaço de busca é definido por

$$p_{rand} = \frac{1}{1 + k\epsilon}, \tag{4.7}$$

em que k representa o número de nós da RRT e o parâmetro ϵ define o quão rápido esta probabilidade decai. Então, a probabilidade de selecionar o objetivo é dada por

$$p_{goal} = 1 - p_{rand}. \tag{4.8}$$

Desta forma, a probabilidade de selecionar um estado aleatório no início do algoritmo é aproximadamente 1 e conforme o número de nós da árvore vai aumentando, a probabilidade de selecionar um estado aleatório vai diminuindo com taxa de decaimento parametrizada por ϵ.

A próxima etapa consistiu na implementação da métrica de distância para a definição do vizinho mais próximo.

4.6 Obtenção do vizinho mais próximo

Nesta etapa, foi necessário definir qual métrica de distância seria utilizada para o crescimento da RRT. Para tal, foi feito um estudo a respeito dos efeitos da métrica no desempenho do algoritmo. De acordo com Cheng e LaValle (2001), a sensibilidade do desempenho da RRT com relação à métrica escolhida é uma das maiores desvantagens desta técnica. Para sistemas dinâmicos, a métrica aplicada pode fornecer informações ilusórias de proximidade que podem aumentar dramaticamente o tempo de solução do algoritmo.

Para ilustrar este efeito da métrica de distância, é utilizado o exemplo de planejamento de movimento de um pêndulo com torque limitado utilizando RRT. Na Figura 11, é possível ver três pêndulos com setas que representam suas respectivas velocidades. O pêndulo vermelho representa um estado sorteado q_{rand} por uma RRT. Considerando os pêndulos azul e verde, é possível ver que o verde está mais próximo do pêndulo vermelho quando considerada a métrica de distância Euclidiana. No entanto, devido ao limite de torque, é muito mais difícil para o estado representado pelo pêndulo verde atingir q_{rand} quando comparado ao pêndulo azul. Portanto, em sistemas dinâmicos, o critério de vizinhança considerando apenas a proximidade pode não ser um critério bom o suficiente. Uma solução é levar em consideração não só a posição mas também a velocidade do sistema.

Perez et al. (2012) propõem uma métrica de distância que se baseia no custo fornecido pelo LQR. Como esta métrica é obtida com o conhecimento da dinâmica do sistema, é possível medir a vizinhança de acordo o menor esforço para atingir um estado

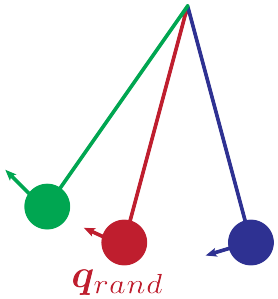


Figura 11 – Ilustração dos efeitos de métricas de vizinhança diferentes na RRT utilizando pêndulos. Apesar do pêndulo verde estar mais próximo, é mais fácil para o pêndulo azul atingir o estado representado pelo pêndulo vermelho.

desejado. Na Figura 11, o vizinho mais próximo de acordo com a métrica LQR seria então o pêndulo azul, em que é possível ver que ele está se movimentando em direção ao pêndulo vermelho, resultando em pouco esforço de controle para atingir o ponto desejado.

Este comportamento ilustra bem a sensibilidade da RRT com a métrica escolhida. Cheng e LaValle (2001) afirmam que a melhor distância é o *cost-to-go* ótimo, que corresponde ao melhor custo para levar o estado de um sistema para um ponto desejado. No entanto, calcular o *cost-to-go* ótimo é um problema tão complexo quanto o problema de geração de trajetória. Por este motivo, custos fornecidos por reguladores lineares quadráticos vêm sendo utilizados (GLASSMAN; TEDRAKE, 2010).

Para a utilização do custo fornecido pelo LQR, o sistema deve ser linearizado e é utilizada a matriz de custo calculada durante a obtenção dos ganhos do regulador como matriz de pesos para ponderar os estados no cálculo da distância Euclidiana. Como um dos critérios definidos para a implementação do planejador de movimentos foi evitar a linearização do sistema, a solução foi definir manualmente uma matriz de pesos e calcular a distância Euclidiana ponderada. Desta forma, é possível dar mais peso a certos estados do sistema no cálculo do vizinho mais próximo, por exemplo dar mais importância à velocidade que à posição do sistema.

Esta abordagem não soluciona o problema da distância Euclidiana não representar bem a dinâmica do sistema, mas reduz a sensibilidade do algoritmo com uma escolha adequada de pesos. Esta abordagem é utilizada por LaValle e Jr (2001), em que são definidos pesos para todos os estados em seus respectivos *manifolds*.

Como o espaço de busca do algoritmo proposto consiste apenas na posição e na velocidade da carga, não seria possível considerar o erro de orientação do quadricóptero nem da carga na métrica de distância, já que o *setpoint* amostrado, dado por (4.6), não contém tal informação. Sendo assim, a métrica de distância usada é definida por

$$d_e^2 = \begin{bmatrix} e_x^T & e_v^T \end{bmatrix} K_w \begin{bmatrix} e_x \\ e_v \end{bmatrix}, \tag{4.9}$$

em que e_x, e_v são os erros de posição e velocidade entre o nó selecionado da arvore e o *setpoint* amostrado, e K_w é uma matriz diagonal de ponderação. A partir desta métrica, o vizinho mais próximo é encontrado e denominado q_{near}.

4.7 Expansão da árvore

Após encontrar o nó mais próximo, a árvore é expandida em direção ao *setpoint* sorteado. Para tal, a dinâmica do sistema é integrada utilizando Runge-Kutta de quarta ordem e o sistema é atuado utilizando o controlador geométrico apresentado na Seção 3.3. A integração é feita durante um intervalo Δt pré-definido. É importante salientar que o intervalo em que ocorre a integração pode ser diferente do passo de integração utilizado no Runge-Kutta, que é comumente representado por h. O primeiro deve ser escolhido a depender do quanto se deseja que o sistema evolua de um nó para o outro e o último deve ser selecionado de forma a manter a estabilidade da simulação e serve como limite inferior para a seleção de Δt.

Esta etapa deve lidar com dois problemas que limitam a exploração e degradam o desempenho da RRT devido à falta de uma métrica ideal que são discutidos por Cheng e LaValle (2001). Isto ocorre porque alguns nós são repetidamente selecionados para expansão da árvore sem realizar progresso na exploração.

O primeiro problema se deve ao fato de que o ponto mais próximo q_{near} depende apenas da métrica, que contém pouca informação da dinâmica. Assim, na ocorrência de um nó em iminência de colisão e na vizinhança do destino, este nó será selecionado como vizinho mais próximo, caso o destino tenha sido amostrado, e poderá gerar nós que colidem e não são adicionados à árvore. Nesta situação, é possível que o nó, que se encontra em iminência de colisão, seja selecionado múltiplas vezes e sempre gere colisões, o que limita o crescimento da árvore.

Este problema foi tratado ao criar duas regras para desabilitar nós da árvores e evitar que sejam candidatos para q_{near}. As regras são:

1. Qualquer nó que cause uma colisão é desabilitado;

2. Qualquer nó que se expanda para um nó já desabilitado também é desabilitado.

Foi necessário adicionar a segunda regra, pois ao desabilitar nós que consistentemente geram colisões, os nós que os geraram passavam a ser candidatos em potencial para q_{near}. Como consequência, havia uma possibilidade de surgir um novo nó no mesmo local que o nó já desabilitado. Na ocorrência de um ramo que gera bastante colisões, há um efeito em cadeia que pode eliminar todos os nós neste ramo. Porém, como a RRT continua se expandindo e gerando novos ramos, o efeito em cadeia não atrapalha o crescimento da árvore.

O segundo problema ocorre quando o sistema de controle atua em q_{near} gerando um ponto mais distante com relação ao ponto amostrado. Isto pode ocorrer por diversos motivos, principalmente em sistemas com bastante inércia. Neste caso, q_{near} irá se manter como um vizinho em potencial nas iterações futuras gerando sempre o mesmo ponto, o que trava o crescimento da árvore. Este problema é relevante quando há uma probabilidade de seleção do destino final. Neste caso, há chances do destino ser selecionado várias vezes consecutivas.

Este problema é resolvido mantendo o mesmo *setpoint* amostrado e continuando a integração toda vez que o nó expandido estiver no mesmo local de um nó que já se encontra na árvore. Este procedimento se repetirá até que surja um q_{new} mais próximo do *setpoint* amostrado, evitando que a árvore se expanda constantemente para o mesmo nó. Este procedimento pode ser realizado porque o controlador é capaz de levar o sistema para a posição desejada.

4.8 Detecção de colisão

Após a obtenção da configuração q_{new}, é feita uma checagem se durante a movimentação de q_{near} para q_{new} não há colisões. Checar colisões desta forma é custoso e nem sempre fácil, o que afeta a velocidade do algoritmo. Assim, foi feita uma simplificação. Considerando que o intervalo de tempo entre dois nós da RRT (Δt) seja pequeno o suficiente para que o sistema não seja capaz de atravessar obstáculos, basta checar se q_{new} representa uma configuração em que há colisão. O que é uma tarefa simples, considerando que basta checar as extremidades do quadricóptero e da carga.

Caso o valor de Δt não seja pequeno o suficiente, ou os obstáculos possuam espessura muito fina, a checagem de colisão pode falhar, porque, entre um estado e outro, o sistema pode atravessar um obstáculo.

4.9 Planejador de movimento completo

Por fim, o planejador de movimento completo é apresentado no Algoritmo 3. Neste algoritmo, as variáveis representadas com o símbolo $\mathbf{x}$ indicam vetores contendo a posição e velocidade da carga, dada por (4.6), e as variáveis com o símbolo $\boldsymbol{q}$ representam o estado do sistema. Como os elementos de $\mathbf{x}$ pertencem ao vetor $\boldsymbol{q}$, é sempre possível obter $\mathbf{x}$ de $\boldsymbol{q}$. Além disso, a guinada do quadricóptero, apesar de fazer parte do *setpoint* do controlador, não foi considerada no espaço de busca porque é mantida constante, já que não afeta a capacidade de execução da tarefa.

Algoritmo 3: $\mathcal{T} \leftarrow GCRRT(q_{init}, \mathrm{x}_{goal}, N, \Delta t, \epsilon, \rho, K_w)$

```
T ← InicializarÁrvore(q_init);
para j = 1 até N faça
    x_d^rand ← sorteiaSetpoint(x_goal, ε);
    [i_near, q_near] ← VizinhoMaisPróximo(x_d^rand, K_w, T);
    q_new ← NovoEstadoGC(T, i_near, x_d^rand, Δt);
    se LivreDeObstáculo(x_near, x_new) então
        T ← InserirNó(i_near, q_new, T);
        se distância(x_new, x_goal) < ρ então
            retorna T
        fim
    fim
fim
```

O Algoritmo 3 funciona da seguinte maneira. A árvore é inicializada com a configuração inicial do sistema e então entra no laço principal do algoritmo. Neste laço, o *setpoint* do controlador é sorteado com uma probabilidade de selecionar um valor aleatório ou o destino, de acordo com as Seções 4.4 e 4.5. Então, é encontrado o vizinho mais próximo q_{near} considerando a matriz de peso K_w, conforme a Seção 4.6. Em seguida, q_{near} é utilizado como ponto de partida para a obtenção do novo nó q_{new} a partir da integração do modelo matemático por um tempo Δt. Então, é checado se q_{new} colide com algum obstáculo. Caso haja colisão, os procedimentos explicados na Seção 4.7 são executados. Caso contrário, o nó é adicionado na RRT e a busca continua até que o número máximo de nós seja atingido ou x_{new}, que corresponde à apenas aos elementos de posição e velocidade de q_{new}, estejam próximos o suficiente do valor desejado de acordo com o parâmetro ρ.

Os parâmetros iniciais do algoritmo são: o estado inicial do sistema $\boldsymbol{q}_{init}$; o *setpoint* final da carga $\mathbf{x}_{goal}$; o número máximo de iterações N; o intervalo de integração Δt; o fator de exploração ϵ; o parâmetro de parada do algoritmo ρ; e a matriz diagonal de pesos K_w.

Ao atingir a posição final desejada, a árvore pode ser utilizada para se obter a trajetória no espaço de estados, bem como as atuações necessárias para a execução do movimento. O algoritmo proposto foi denominado GCRRT (*Geometric Control RRT*).

5 Resultados e Discussão

Com o objetivo de verificar as capacidades do planejador de movimento proposto, foram criados 4 ambientes com diferentes obstáculos. Todos os testes foram realizados por simulação e foi considerado que todos os estados do modelo são conhecidos. A discussão é feita considerando a viabilidade da trajetória a partir dos esforços de controle.

Em todos os ambientes a carga parte da posição $\boldsymbol{x}_L = [0\ \ -1{,}5\ 4{,}5]$ com destino final $\boldsymbol{x}_{goal} = [0\ 1{,}5\ 4{,}5]$. As posições amostradas no espaço de busca, onde todos os pontos sorteados devem estar contidos, corresponde ao que está sendo mostrado na Figura 12. Com relação às velocidades amostrada, foi utilizado um limite máximo e e mínimo de $2\,\mathrm{m\,s^{-1}}$ e $-2\,\mathrm{m\,s^{-1}}$, respectivamente. Estes parâmetros foram mantidos para todos os ambientes.

Com o intuito de avaliar o algoritmo proposto, foram realizadas 100 execuções do algoritmo em cada ambiente e foram registradas diversas informações como taxa de acerto, número de nós da árvore, tempo de solução e tempo de execução da trajetória. Estas informações estão dispostas neste capítulo e concernem apenas as execuções em que a RRT obteve sucesso na geração de uma trajetória antes de atingir o limite máximo de nós. Este procedimento foi feito em duas versões da RRT proposta. Uma versão do algoritmo não contém a implementação das regras de exclusão de nós apresentada na Seção 4.7 e a segunda versão contém. Com isso, foi possível analisar o efeito da exclusão de nós na taxa de sucesso do algoritmo.

Com relação aos parâmetros do algoritmo proposto, $N = 25000$, $\Delta t = 0{,}05$, $\epsilon = 1 \cdot 10^{-5}$, $\rho = 5 \cdot 10^{-1}$, $K_w = diag(1; 1; 1; 1 \cdot 10^{-2}; 1 \cdot 10^{-2}; 1 \cdot 10^{-2})$.

Os parâmetros do controlador geométrico estão apresentados na Tabela 5. Estes valores estão diferentes dos ganhos dispostos na Tabela 4, que foram utilizados apenas para verificar o funcionamento do controlador, porque os ambientes testados necessitam de movimentos mais agressivos que podem ser alcançados com o aumento no ganho do controlador.

Todos os experimentos foram realizados em um MacBook Pro (Early 2011) com processador Intel Core i7–2620M (4MB cache) de segunda geração, 4GB of DDR3 (1333 Hz), 250GB SSD, e Matlab R2014a.

Adicionalmente, foi postado em um repositório de vídeos algumas das RRTs geradas e trajetórias obtidas nos experimentos com todos os ambientes testados para facilitar a análise do algoritmo proposto. O vídeo pode ser acessado em https://youtu.be/afqn9CpTXaw.

Tabela 5 – Ganhos do controlador geométrico.

Constante	**Valor**
k_R	$diag(1,40; 1,40; 0,40)$
k_Ω	$diag(0,12; 0,12; 0,08)$
k_q	13
k_ω	9,5
k_x	$diag(9,00; 9,00; 9,75)$
k_v	$diag(6,90; 6,90; 5,175)$

$diag(.;.;.)$ especifica os elementos de uma matriz diagonal.

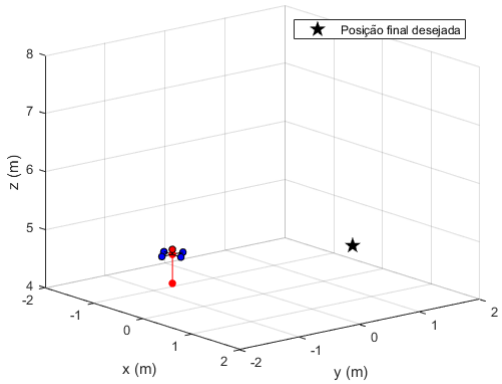


Figura 12 – Ambiente 1, sem obstáculos.

5.1 Ambientes de teste do algoritmo proposto

São propostos quatro ambientes para verificar as capacidades da RRT proposta.

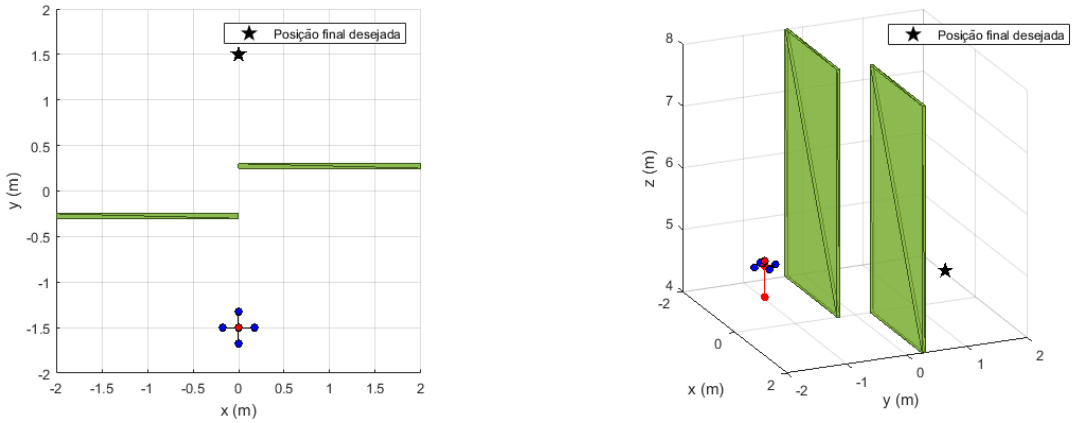
5.1.1 Ambiente 1

O primeiro ambiente de teste, ilustrado na Figura 12, não contém obstáculos e foi escolhido para servir de padrão com o intuito de avaliar melhor a capacidade de exploração do espaço de busca do algoritmo em ambientes mais complexos.

5.1.2 Ambiente 2

O Ambiente 2, ilustrado na Figura 13, foi projetado pensando na possibilidade de obstáculos simples, de forma que seja possível transpor com movimentos suaves ou agressivos. Neste ambiente, a distância entre os dois obstáculos é de 0,5 m, enquanto a envergadura do quadricóptero é de 0,35 m, ou seja, o quadricóptero tem espaço suficiente para atravessar sem a necessidade de manobras complexas. Para evitar que a RRT tente

encontrar caminhos em torno dos obstáculos e permitir uma abertura razoável a fim de reduzir o tempo de processamento da RRT no problema, é considerado que as paredes possuem comprimento infinito fora do espaço ilustrado nas imagens. O mesmo também ocorre para os Ambientes 3 e 4.

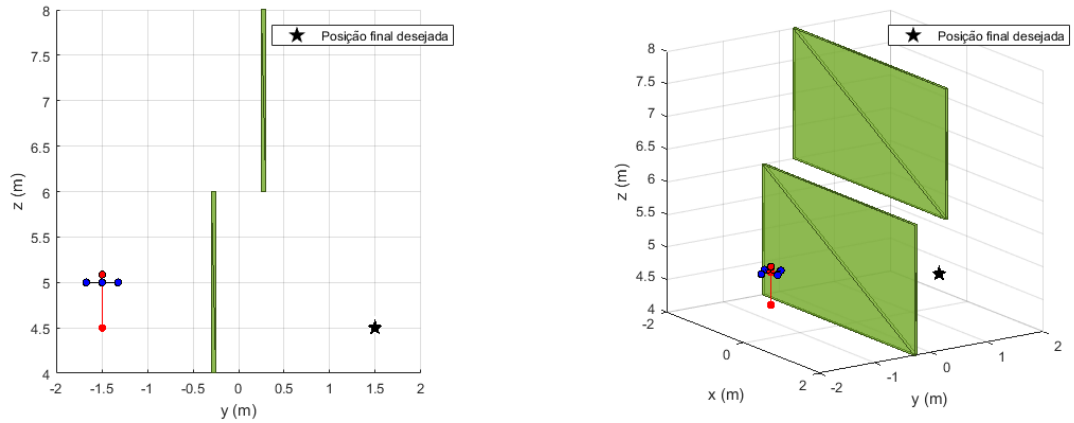


(a) (b)

Figura 13 – Ambiente 2: (a) vista superior e (b) vista panorâmica. Este ambiente possui obstáculos que exigem movimentação no plano xy.

5.1.3 Ambiente 3

O Ambiente 3 foi projetado com o intuito de avaliar a movimentação do sistema no eixo z, pois a dinâmica de movimentação neste eixo é bem diferente dos outros. Assim, este ambiente foi obtido ao rotacionar o Ambiente 2 por 90° em torno do eixo x, como pode ser visto na Figura 14.

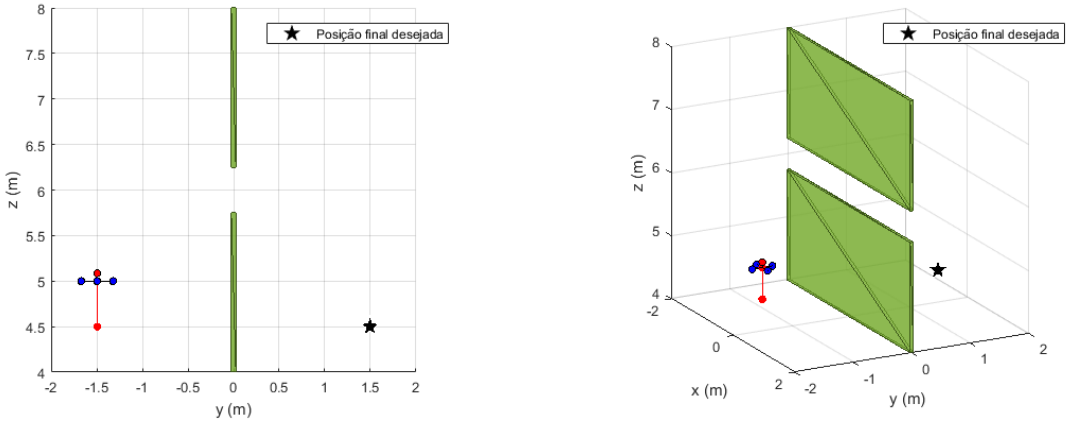


(a) (b)

Figura 14 – Ambiente 3: (a) vista lateral e (b) vista panorâmica. Este ambiente possui obstáculos que exigem movimentação no plano zy.

5.1.4 Ambiente 4

O Ambiente 4 corresponde ao problema da janela curta, em que o quadricóptero com carga suspensa deve atravessar dois obstáculos cuja distância entre eles é menor que a altura do sistema. A altura total do sistema corresponde a 0,5855 m enquanto a altura da janela é de 0,5 m, que é em torno de 85% do comprimento total do sistema. O ambiente está ilustrado na Figura 15.



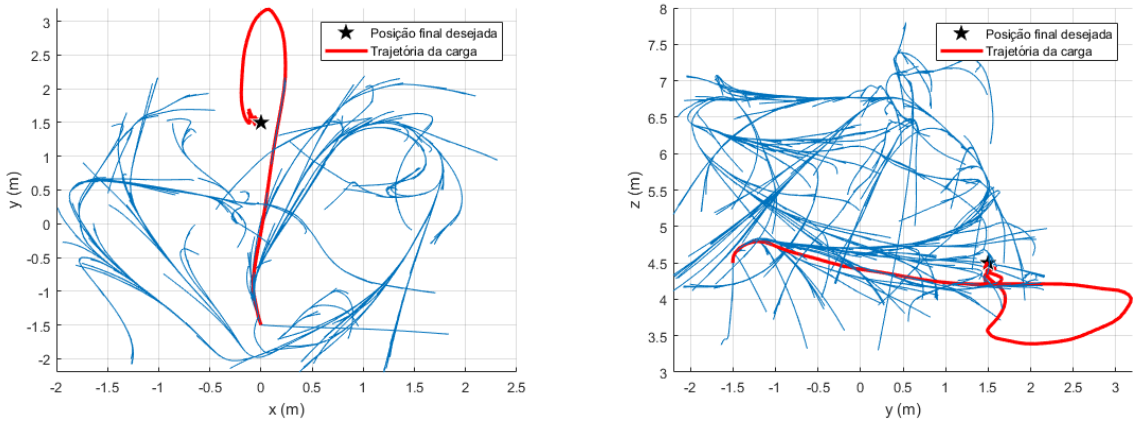
(a) (b)

Figura 15 – Ambiente 4: (a) vista lateral e (b) vista panorâmica. Este ambiente possui obstáculos cuja abertura é 85% do comprimento do sistema quadricóptero-carga:

5.2 Experimentos com o Ambiente 1

Resolver o Ambiente 1 é um problema trivial no ponto de vista de controle. Bastaria colocar o objetivo como *setpoint* que o controlador encontraria uma trajetória realizável. Apesar disso, este ambiente serviu para verificar a capacidade de busca da RRT. Dos 100 testes realizados, houve 100% de sucesso em encontrar um caminho factível. Como neste ambiente não há obstáculos, ambas as versões do algoritmo atuam da mesma maneira.

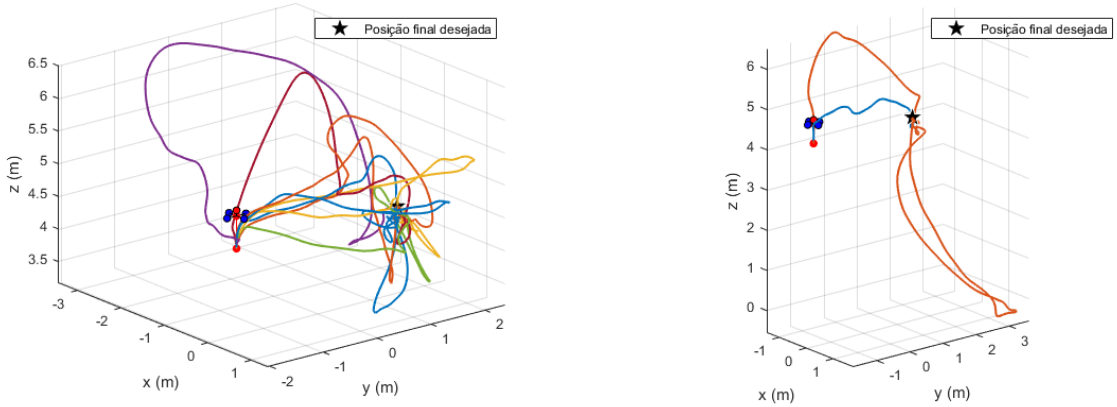
A RRT gerada em um dos testes está ilustrada na Figura 16. É possível ver que a RRT explorou relativamente bem o ambiente em torno da posição final desejada. Mesmo assim, é possível ver também que vários ramos da RRT ficaram concentrados ao invés de se espalhar no ambiente de busca. Este efeito é um resultado de dois aspetos: o problema da métrica euclidiana que seleciona o mesmo nó diversas vezes e a abordagem de continuar integrando um nó que já foi selecionado. É importante ressaltar que continuar integrando um nó que já possui um nó filho foi uma decisão implementada para evitar que alguns ramos da RRT continuassem gerando o mesmo nó e, consequentemente, travassem seu crescimento. Este procedimento está explicado com mais detalhes na Seção 4.7.



(a) (b)

Figura 16 – RRT, projetada no espaço tridimensional, gerada no Ambiente 1: (a) vista superior e (b) vista lateral.

Com o intuito de ilustrar os caminhos obtidos nos diferentes testes, são apresentados os caminhos gerados em 10 dos 100 experimentos na Figura 17a e, na Figura 17b, a trajetória mais curta e a mais longa considerando os 100 experimentos. As trajetórias obtidas pelo algoritmo foram bastante diferentes devido à natureza aleatória do algoritmo, obtendo umas mais longas e outras mais curtas.



(a) (b)

Figura 17 – Trajetórias geradas pela RRT proposta para o Ambiente 1: (a) contém 10 trajetórias entre as 100 obtidas e (b) contém a trajetória mais curta e a mais longa.

A trajetória mais curta foi relativamente direto ao destino e com pouca sobrepassagem. Na trajetória mais longa, vê-se que, em algum momento, o sistema se aproximou bastante do destino final mas não concluiu a movimentação. O algoritmo não considerou como finalizado, porque a condição de parada também considera as velocidades do sistema. Assim, a RRT continuou crescendo e explorando o ambiente até se aproximar do objetivo e do ponto de equilíbrio. Adicionalmente, foi verificado que esta sobrepassagem, de cerca

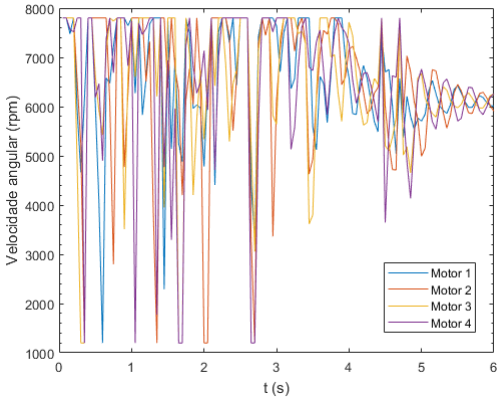


Figura 18 – Esforço de controle dos motores referente à trajetória mais curta obtida pela RRT no Ambiente 1.

de 6 m no eixo z, ocorreu porque o quadricóptero executou uma manobra muito agressiva na tentativa de cancelar as oscilações e acabou tendo que se afastar do destino para re-estabilizar o sistema.

Para evidenciar que a solução encontrada satisfaz a restrição de velocidade angular dos motores, os esforços de cada motor referentes à trajetória mais curta são mostrados na Figura 18. É possível ver que as velocidades dos motores estão sempre entre os valores mínimos e máximos permitidos, que correspondem respectivamente a 1200 rpm e 7800 rpm.

Por fim, são apresentados na Tabela 6, os valores médios, máximos e mínimos relativos ao número de nós gerados na RRT, ao tempo de solução do algoritmo e ao tempo de execução da trajetória. O número de nós gerados está de acordo com soluções obtidas na literatura para sistemas com alta dimensão, porém o tempo de execução está maior. No entanto, é possível ver na Tabela 6 que o valor máximo do tempo de solução foi cerca de 640 minutos. Este tempo ocorreu apenas uma vez, o que configura um ponto fora da curva. Para ilustrar melhor os tempos de solução, um histograma é apresentado na Figura 19. Nesta imagem, o tempo máximo é excluído para facilitar a visualização dos outros tempos. Vê-se que a maior parte das soluções levam entre 5 a 10 minutos para encontrar um caminho.

A demora para encontrar uma solução ocorre devido à utilização de uma linguagem de programação interpretada, enquanto na literatura é utilizada uma linguagem compilada, como o C. Outro motivo para a demora da solução é a quantidade de integrações que ocorre em um mesmo nó devido à sensibilidade do algoritmo com a métrica Euclidiana, que acaba selecionando o mesmo vizinho mais próximo por diversas vezes.

Adicionalmente, o tempo de execução da trajetória também é apresentado para servir de base para os outros experimentos. Os valores obtidos de tempo de execução da trajetória indicam que as velocidades resultantes foram bastante lentas, considerando que

Tabela 6 – Dados das soluções no Ambiente 1.

Característica	**Valor médio**	**Valor máximo**	**Valor mínimo**
Número de nós	7983	22525	4374
Tempo da solução	20,2 min	640,7 min	4,0 min
Tempo da trajetória	10,4 s	18,9 s	6,1 s

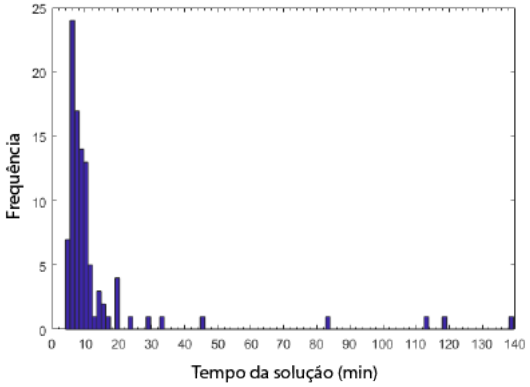


Figura 19 – Histograma do tempo de solução para o Ambiente 1.

levou em média 10 s para navegar apenas 3 m. Isto se deve porque o quadricóptero deve reduzir bastante a velocidade quando se aproxima do objetivo para que a carga pare de oscilar e atinja o ponto de equilíbrio na posição desejada.

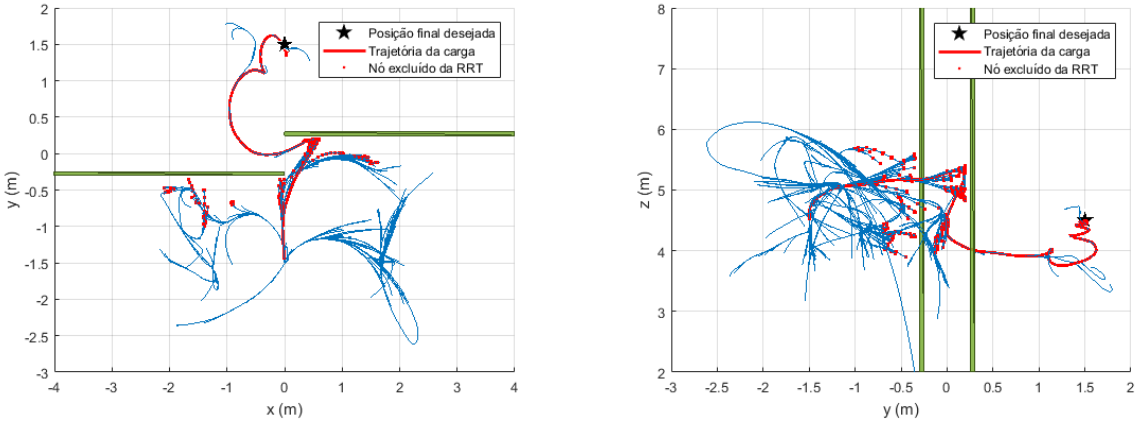
Assim que a RRT conecta o destino à configuração desejada, o algoritmo retorna uma trajetória discretizada que contém todos os estados do sistema quadricóptero-carga e também todas as atuações necessárias para atingir os estados da trajetória.

5.3 Experimentos com o Ambiente 2

Para solucionar o Ambiente 2, o quadricóptero precisa fazer uma curva no plano xy para evitar colisão com os obstáculos. Como o espaço livre entre os obstáculos é maior que a envergadura do quadricóptero, não há a necessidade de manobras agressivas para transpor o obstáculo. A dificuldade está em controlar a oscilação da carga para evitar colisões durante a curva.

Ao analisar a Figura 20, em que é apresentada a expansão da árvore de uma possível solução, é possível ver foram gerados vários nós que colidiram e foram excluídos da árvore até que um ramo que transpõe o obstáculos fosse encontrado. É possível ver também que a RRT não se espalhou tanto quanto no Ambiente 1. Este fato pode indicar que durante o crescimento da árvore poucos nós estão sendo selecionados múltiplas vezes, o que diminui a quantidade de ramos da RRT. Além disso, quando estes nós colidem seus parentes continuam sendo escolhidos e levando a mais colisões, que também trava o

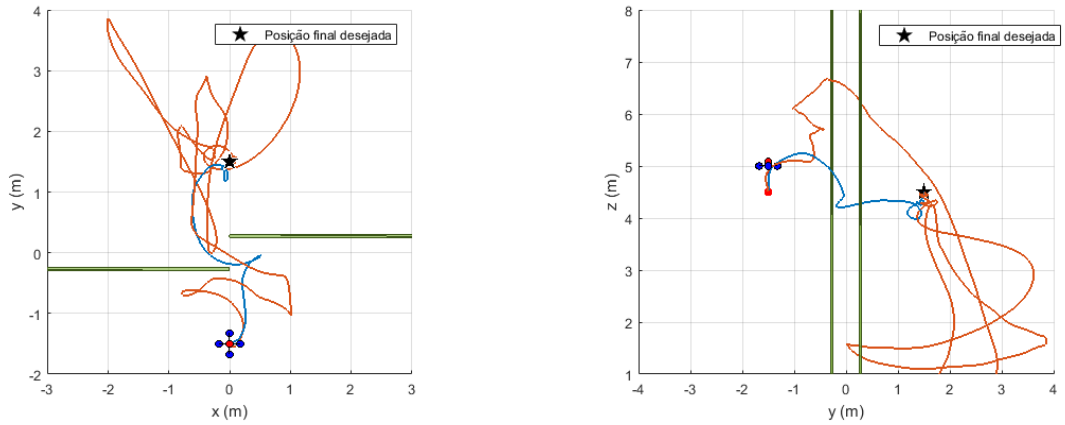
crescimento da RRT. Este comportamento pode ser visto nas Figuras 20a e 20b onde há rastros de nós excluídos, indicando que o mesmo ramo continuou tentando se expandir múltiplas vezes em um caminho que levava a colisões.



(a) (b)

Figura 20 – RRT, projetada no espaço tridimensional, gerada no Ambiente 2: (a) vista superior e (b) vista lateral.

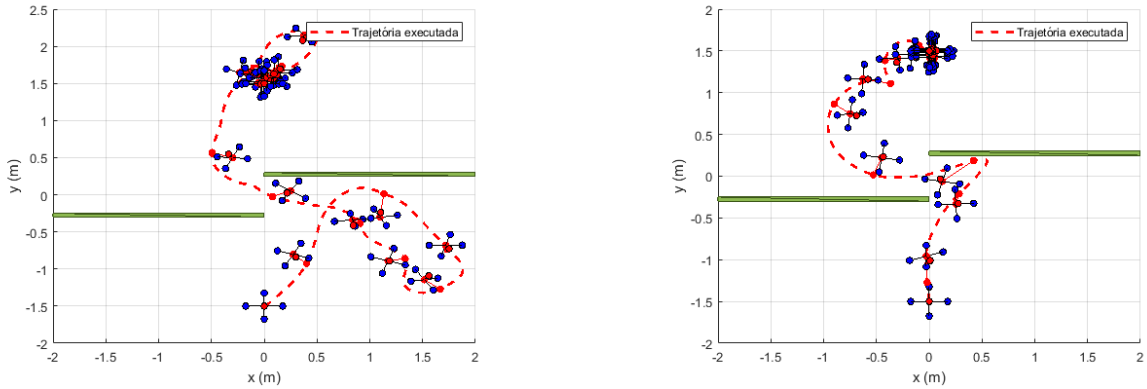
Devido a este comportamento, dos 100 testes realizados, a taxa de sucesso do algoritmo foi de 27% na versão com exclusão de nós e 14% na versão sem exclusão. A trajetória mais curta e a mais longa encontradas considerando a exclusão de nós estão apresentadas na Figura 21. É possível ver, na Figura 21a, que as trajetórias realmente evitam os obstáculos com uma distância de segurança de 5 cm nos eixos x e y. Este intervalo de segurança foi escolhido com dois propósitos: evitar que o sistema atravessasse o obstáculo durante os passos de integração e fornecer uma segurança na execução da tarefa. Na Figura 21b, vê-se que as soluções da RRT se expandem ao longo do eixo z no espaço de busca.



(a) (b)

Figura 21 – Trajetórias, contendo a mais longa e a mais curta, geradas pela RRT proposta no Ambiente 2: (a) vista lateral e (b) vista superior.

Com o intuito de ilustrar melhor a movimentação do sistema durante a execução das trajetórias, foram selecionadas duas soluções contendo as configurações do sistema quadricóptero-carga ao longo da trajetória. Tais soluções estão dispostas na Figura 22. Vê-se que uma solução, Figura 22a, foi mais conservadora fazendo o quadricóptero dar uma volta para melhor se posicionar para transpor o obstáculo. Por outro lado, a segunda solução seguiu uma trajetória mais arriscada ao executar uma curva muito mais fechada. Inclusive, a árvore apresentada na Figura 20 contém a solução do segundo caso.



(a) (b)

Figura 22 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 2.

Adicionalmente, as velocidade dos motores executadas na trajetória da Figura 22a estão apresentadas na Figura 23. É possível ver que as restrições dos motores foram respeitadas em todos os momentos.

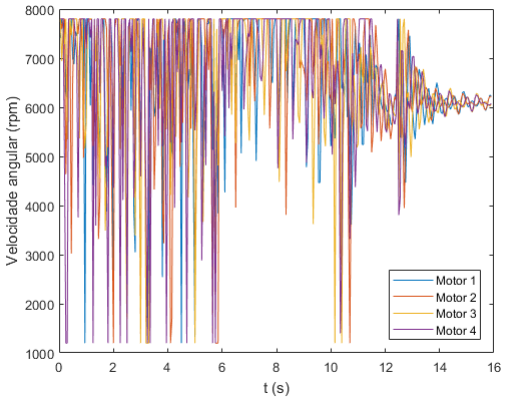


Figura 23 – Velocidades dos motores executadas na solução da Figura 22a.

Por fim, os dados obtidos com relação à RRT com exclusão de nós no Ambiente 2 estão apresentados na Tabela 7. Com relação ao Ambiente 1, houve um aumento no número médio de nós gerados na RRT, que já era esperado considerando que a árvore continua crescendo até consiga transpor o obstáculo. Em contrapartida, o tempo médio da solução diminuiu. Isto pode ter ocorrido devido a um desvio da média causado pelo

ponto fora da curva. Ao avaliar o histograma apresentado na Figura 24, é possível ver que as soluções estão distribuídas em tempos maiores que 10 minutos, indicando que neste experimento as soluções demoraram mais para serem obtidas que no Ambiente 1. Além disso, como o sistema tem que executar manobras para transpor o obstáculo, o tempo médio de execução da trajetória também aumentou.

Tabela 7 – Dados das soluções com exclusão de nós no Ambiente 2.

Característica	**Valor médio**	**Valor máximo**	**Valor mínimo**
Número de nós	14746	23871	8264
Tempo da solução	19,0 min	55,1 min	9,7 min
Tempo da trajetória	13,3 s	23,7 s	7,5 s

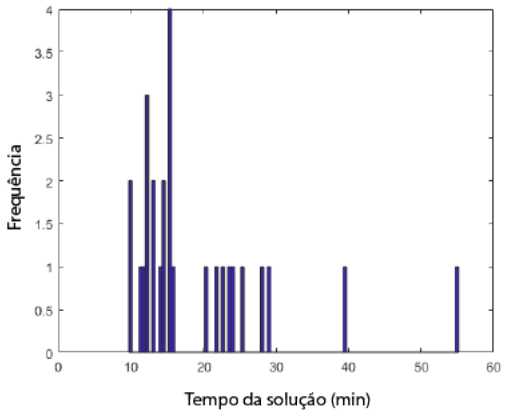


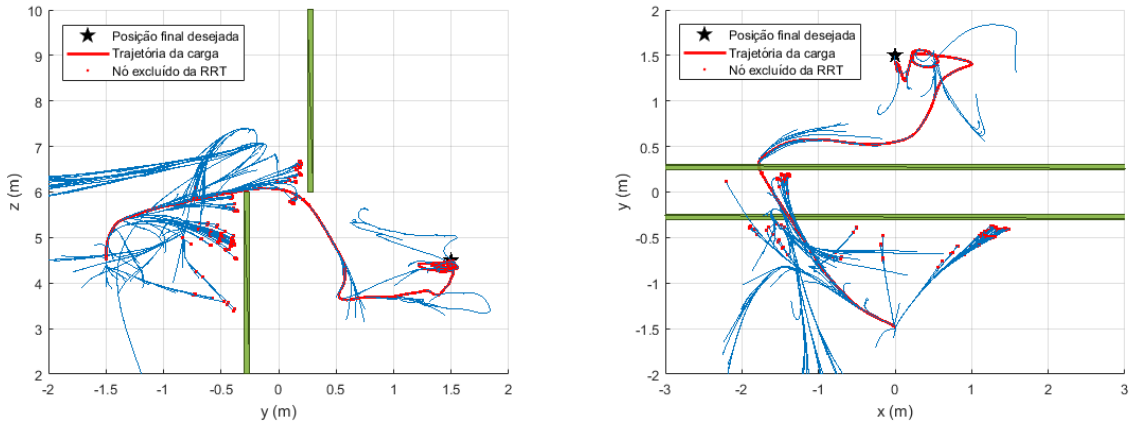
Figura 24 – Histograma do tempo de solução para o Ambiente 2.

5.4 Experimentos com o Ambiente 3

O Ambiente 3 foi criado a partir de uma rotação do Ambiente 2 e a solução para transpor o obstáculo é análoga. Neste caso, o quadricóptero precisa fazer uma curva no plano yz.

A RRT gerada de uma possível solução é apresentada na Figura 25. Vê-se os mesmos problemas que ocorreram na RRT do Ambiente 2: pouca expansão da RRT e constante seleção de nós que levam a colisões.

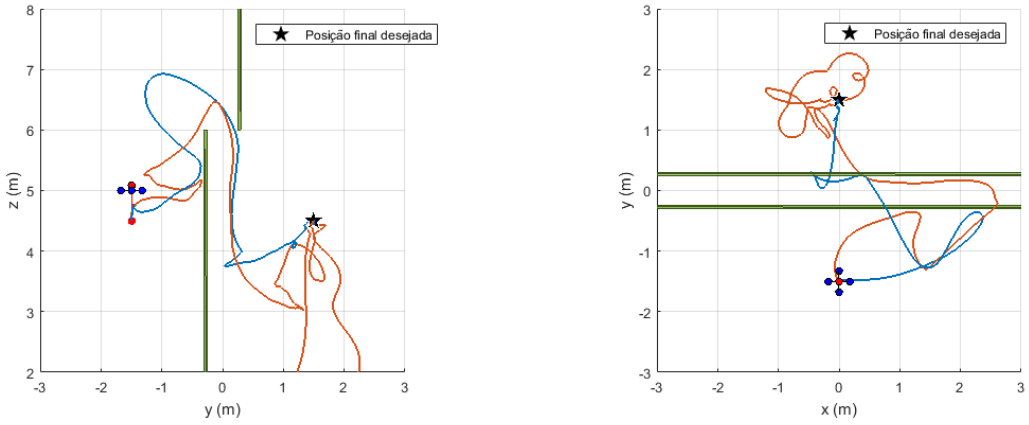
Apesar da pouca expansão da árvore, a taxa sucesso nos 100 experimentos foi de 43% na versão com as regras de exclusão de nós e de 37% na versão sem tais regras. As soluções referentes à trajetória mais curta e a mais longa no algoritmo com as regras de exclusão estão ilustradas na Figura 26. Acredita-se que o aumento na taxa de sucesso



(a) (b)

Figura 25 – RRT, projetada no espaço tridimensional, gerada no Ambiente 3: (a) vista lateral e (b) vista superior.

do algoritmo se deve ao fato de que a curva necessária para transpor o obstáculo ocorre na direção da aceleração da gravidade. Este fato causa menos oscilação da carga quando comparado ao Ambiente 2, em que a manobra é executada perpendicularmente à gravidade.



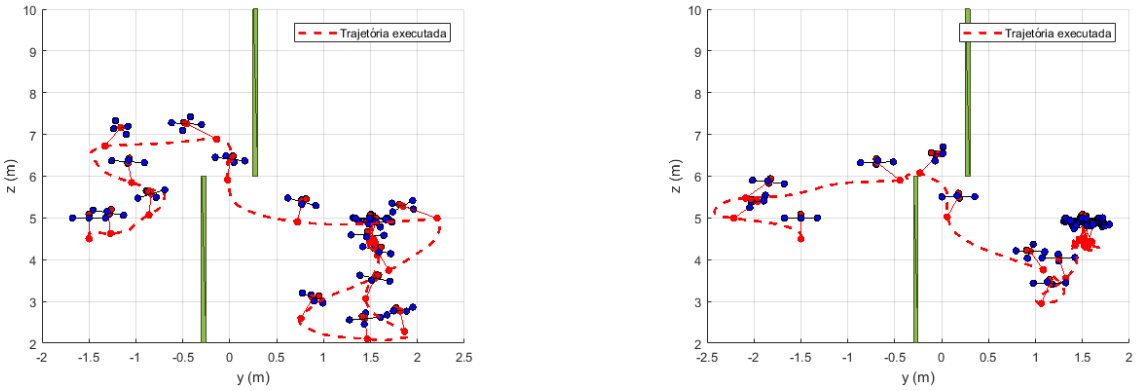
(a) (b)

Figura 26 – Trajetórias, contendo a mais longa e a mais curta, geradas pela RRT proposta no Ambiente 3: (a) vista lateral e (b) vista superior.

Outro fato que pode ter auxiliado no aumento da taxa de sucesso foi a adição da distância de segurança de 5 cm apenas na direção do eixo y. Apesar de ser menos seguro para a carga, a distância de segurança não foi considerada no eixo z, porque, neste sentido de movimentação, o quadricóptero tende a variar menos para os problemas investigados.

É importante salientar também que em um dos experimentos realizados neste ambiente, o sistema atravessou o obstáculo na direção de movimentação do eixo y. Isto ocorreu mesmo com o intervalo de segurança, o que indica que a velocidade do quadricóptero estava muito alta e durante a integração o obstáculo foi ignorado. Neste caso, o sucesso da solução foi desconsiderado.

Na Figura 27, são apresentadas as movimentações ao longo de duas trajetórias encontradas com a RRT proposta. As duas apresentam movimentações agressivas, mas a solução da Figura 27a aparenta ser mais conservadora, considerando que o quadricóptero subiu para depois descer e transpor o obstáculo em segurança. Em contrapartida, a solução da Figura 27b foi mais arriscada passando rente ao obstáculo.



(a) (b)

Figura 27 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 2.

As velocidades dos motores referentes à trajetória 27a estão apresentadas na Figura 28. Vê-se que as velocidades respeitam as restrições dos motores.

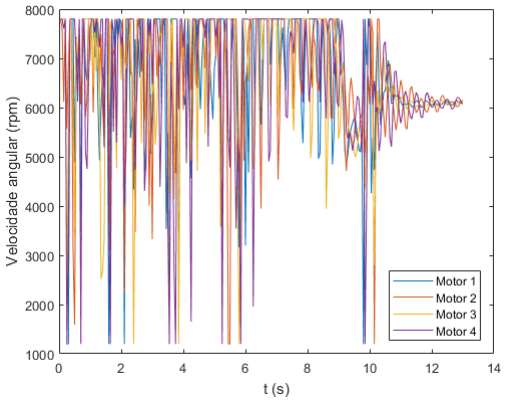


Figura 28 – Velocidades dos motores executadas na solução da Figura 27a.

Por fim, os dados obtidos com relação à RRT com exclusão de nós no Ambiente 3 estão apresentados na Tabela 8. Com relação aos resultados obtidos no Ambiente 1, todos os valores médios aumentaram conforme o esperado, com exceção do tempo médio da solução. Porém, ao avaliar o histograma apresentado na Figura 29, vê-se que a maioria das soluções se encontram ente 10 e 25 minutos. Ao comparar com o Ambiente 2, o algoritmo demorou mais tempo para encontrar as soluções no Ambiente 3.

Tabela 8 – Dados das soluções com exclusão de nós no Ambiente 3.

Característica	**Valor médio**	**Valor máximo**	**Valor mínimo**
Número de nós	15388	23680	8138
Tempo da solução	20,2 min	46,4 min	8,7 min
Tempo da trajetória	14,3 s	21,1 s	10,6 s

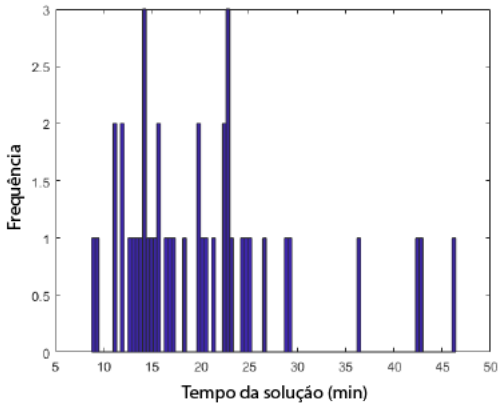


Figura 29 – Histograma do tempo de solução para o Ambiente 3.

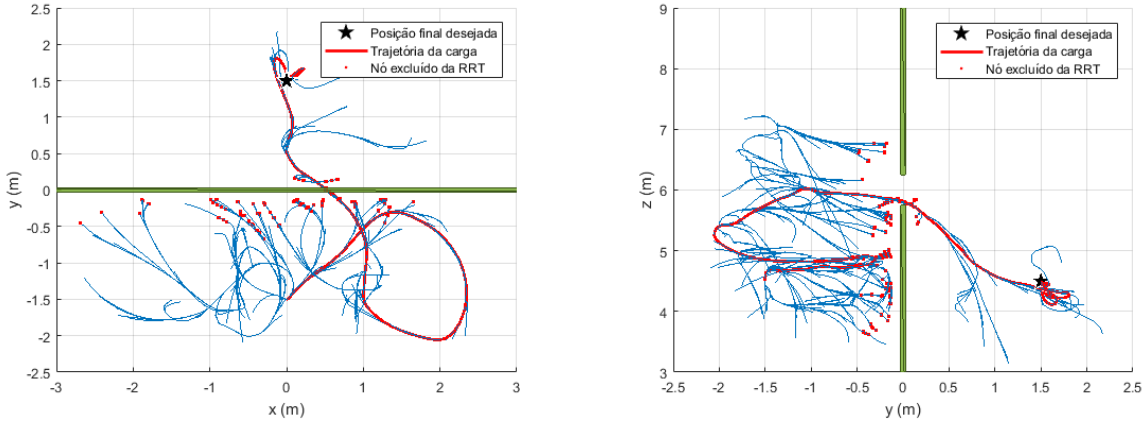
5.5 Experimentos com o Ambiente 4

O Ambiente 4, diferentemente dos outros, exige uma solução em que o quadricóptero precisa executar movimentos mais complexos. Como a janela é mais curta que o comprimento do sistema quadricóptero-carga, o mesmo não é capaz de transpor a janela sem uma manobra que levante a carga para que ambos possam atravessar os obstáculos.

A complexidade da tarefa pode ser entendida ao analisar a Figura 30. Pode ser visto que, apesar da árvore ter se expandido em vários ramos, uma pequena quantidade foi capaz de transpor a janela.

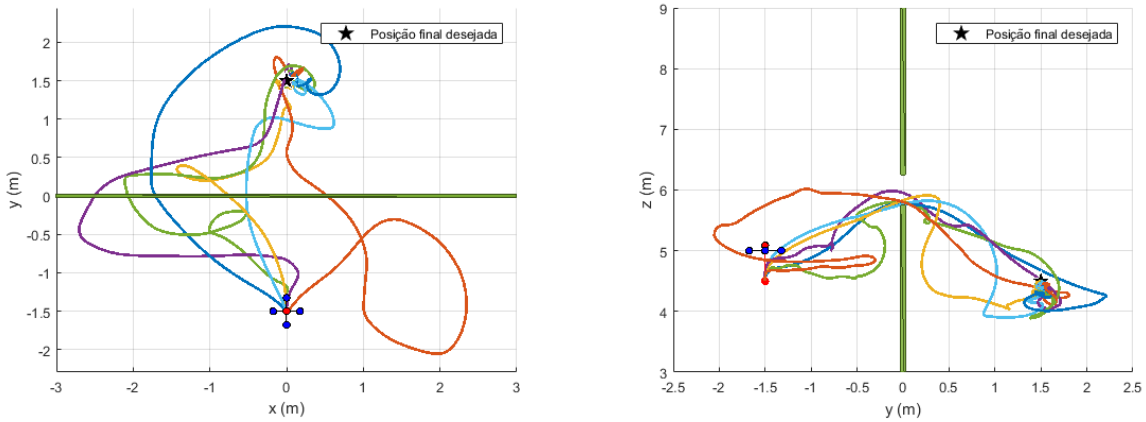
É possível ver que muitos nós colidiram com os obstáculos e foram excluídos da árvore, até mesmo alguns nós que já haviam transposto a janela. Isto ocorreu porque a representação gráfica do nó é feita apenas na posição da carga enquanto a colisão considera o sistema completo. Foi utilizado uma distância de segurança de 10 cm na direção do eixo y para evitar que o sistema atravessasse o obstáculo devido à alta velocidade necessária para transpor este ambiente.

Após a execução de 100 testes no Ambiente 4, foi obtida uma taxa de sucesso de 6% no algoritmo com as regras de exclusão de nós e de 5% na versão que não contém. Todas as trajetórias encontradas que solucionam o problema estão ilustradas na Figura 31.



(a) (b)

Figura 30 – RRT, projetada no espaço tridimensional, gerada no Ambiente 4: (a) vista superior e (b) vista lateral.

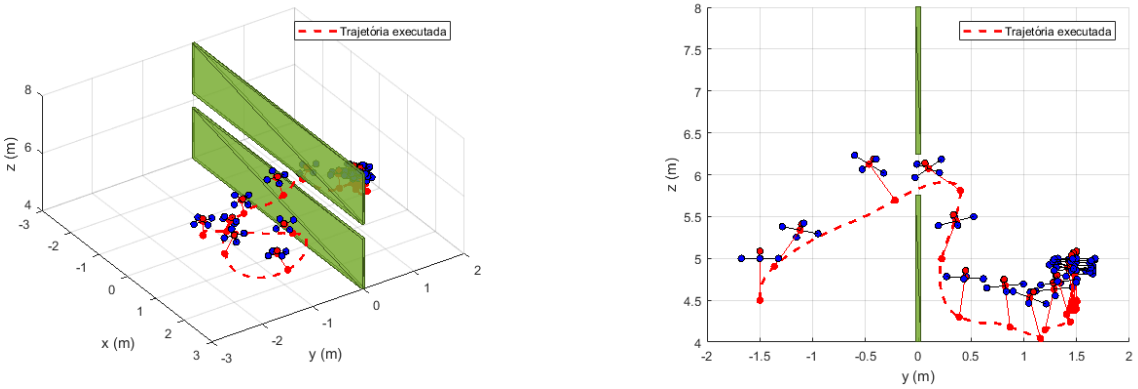


(a) (b)

Figura 31 – Trajetórias geradas pela RRT proposta no Ambiente 4: (a) vista superior e (b) vista lateral.

A essência da solução e a estratégia utilizada para transpor o obstáculo não são reveladas apenas ilustrando a trajetória da carga. Por este motivo, as trajetórias e as configurações espaciais do sistema completo ao longo do tempo estão ilustradas na Figura 32.

Para o caso da solução ilustrada na Figura 32a, é possível ver que o quadricóptero executou um movimento circular na carga que permitiu acumular energia suficiente para que a carga mantivesse a altura necessária para atravessar a janela enquanto o quadricóptero reduzia a altura para atravessar a janela simultaneamente. A solução apresentada na Figura 32b contém uma trajetória mais curta. O sistema foi capaz de transpor o obstáculo porque o quadricóptero partiu com bastante velocidade e arremessou a carga para a frente.



(a) (b)

Figura 32 – Duas possíveis movimentações geradas pela RRT proposta no Ambiente 4.

As outras soluções encontradas envolvem estratégias de movimentação semelhantes às duas apresentadas na Figura 32. No entanto, é possível que existam diversas estratégias diferentes para resolver o problema que não foram obtidas neste experimento.

Com relação aos esforços de controle, as velocidades angulares dos motores executadas na trajetória ilustrada na Figura 32b são apresentados na Figura 33. Durante o primeiro momento da execução (entre 0 s e 1 s), é possível ver que os motores saturam na velocidade máxima por diversas vezes. Este momento condiz com as três primeiras ilustrações do sistema (da esquerda para a direita) na Figura 32b. É possível ver que estão bastante espaçadas, indicando uma alta velocidade neste intervalo. No intervalo entre 1 s e 2 s, os motores saturam diversas vezes no valor mínimo, que é o momento em que a carga é arremessada para frente e o quadricóptero está atravessando a janela.

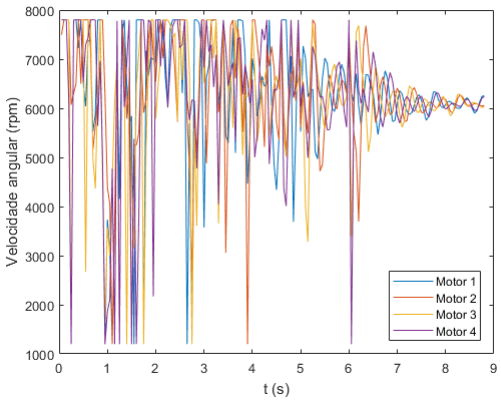


Figura 33 – Esforço de controle dos motores de da solução apresentada na Figura 32b.

Por fim, os dados obtidos com relação à RRT com exclusão de nós no Ambiente 4 estão apresentados na Tabela 9. Em comparação com os dados do Ambiente 1, sem obstáculos, apresentados na Tabela 1, foi possível ver que: o número médio de nós aumentou; o tempo médio da solução diminuiu; e o tempo médio de execução da trajetória foi menor. É difícil afirmar motivos das alterações, principalmente porque só foram encontradas 6

soluções, mas é possível que o número de nós tenha aumentado devido à maior exploração do ambiente para que a árvore pudesse transpor os obstáculos. No entanto, o tempo de solução foi menor, quando era esperado que fosse ser maior justamente pelo aumento no número de nós da árvore. Além disso, é possível ver no histograma apresentado na Figura 34, que houve soluções rápidas, durando 6 minutos, e soluções mais longas, em torno de 25 minutos. Com relação ao tempo de execução da trajetória, é possível que tenha sido menor devido à velocidade que o quadricóptero deve executar a manobra para transpor a janela.

Tabela 9 – Dados das soluções com exclusão de nós no Ambiente 4.

Característica	**Valor médio**	**Valor máximo**	**Valor mínimo**
Número de nós	12504	21803	6305
Tempo da solução	13,6 min	24,9 min	6,0 min
Tempo da trajetória	9,2 s	13,1 s	6,2 s

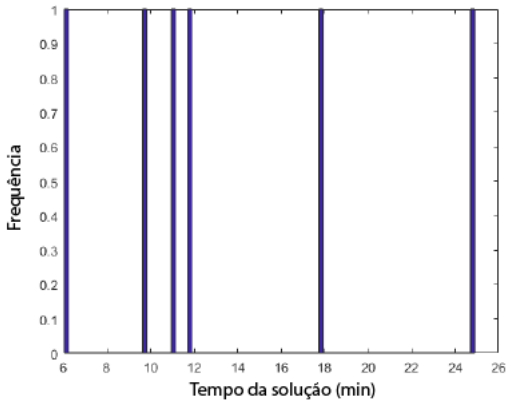


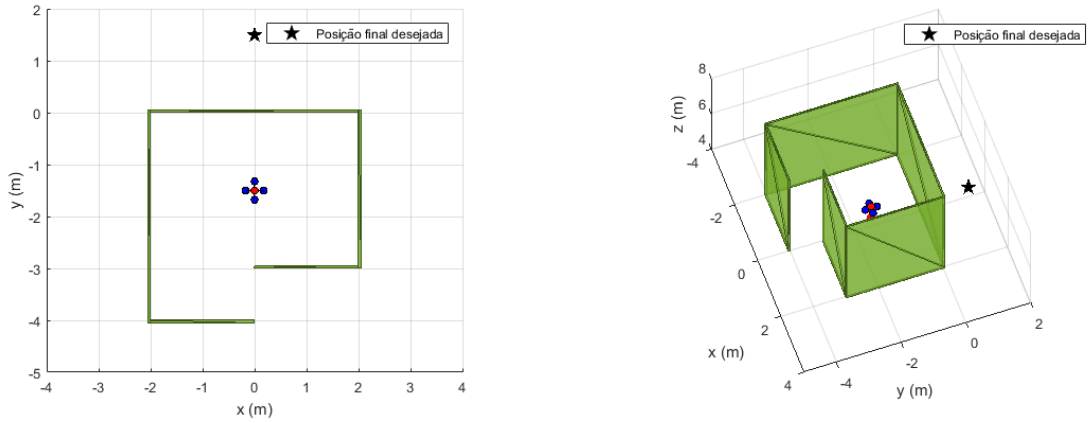
Figura 34 – Histograma do tempo de solução para o Ambiente 4.

5.6 Teste em um ambiente com um obstáculo não convexo

Os experimentos com os Ambientes 1 a 4 demonstram a capacidade do algoritmo proposto em resolver problemas com obstáculos que possam necessitar movimentações agressivas. Durante a realização dos experimentos, a troca do ambiente necessitou apenas da informação das regiões ocupadas pelos sólidos para checar colisões, diferentemente das técnicas na literatura em que deve haver uma preparação específica para cada ambiente, seja através do projeto de uma função de custo ou da escolha de pontos de amostragem para auxiliar na obtenção da trajetória.

Como consequência da independência do formato dos obstáculos, a RRT é capaz de resolver ambientes com obstáculos não convexos, que os planejadores baseados em otimização encontrados na literatura não são, devido à ocorrência de mínimos locais.

Entretanto, encontrar trajetórias em ambientes com obstáculos não convexos não é uma tarefa fácil. Muitas vezes, o obstáculo se encontra entre o sistema e o objetivo portanto, para transpor o obstáculo não-convexo, o algoritmo deve fazer um movimento na direção oposta ao objetivo. Um exemplo de ambiente em que esta situação ocorre está ilustrado na Figura 35. Este ambiente foi projetado para se assemelhar com o Ambiente 2, ilustrado na Figura 13.



(a) (b)

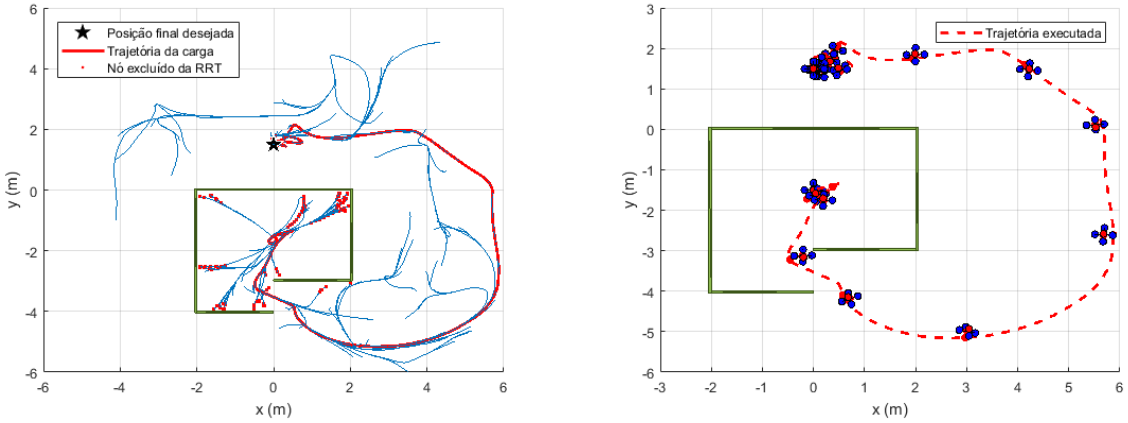
Figura 35 – Ambiente com obstáculo não convexo: (a) vista lateral e (b) vista panorâmica.

A RRT é um algoritmo capaz de explorar o ambiente e evitar o obstáculo mesmo que ele esteja entre a configuração inicial e a final. No entanto, a existência do termo de polarização utilizado na RRT proposta aumenta a probabilidade de sorteio do destino conforme a RRT explora o ambiente. Este comportamento causa um conflito, considerando que a RRT precisa crescer no sentido contrário para transpor o obstáculo mas há um fator forçando a árvore a crescer em direção ao destino. A consequência deste comportamento é uma maior sensibilidade ao parâmetro de polarização ϵ na ocorrência de obstáculos não convexos.

Sabendo desta dificuldade, foi feito um teste em um ambiente com um obstáculo não convexo. Este teste não é representativo, por não ter sido feita uma análise estatística, mas é ilustrativo para avaliar a capacidade do algoritmo proposto em um ambiente com um obstáculo não convexo. Deste modo, este teste não pode ser comparado diretamente com os experimentos feitos nos Ambientes 1 a 4 devido à necessidade de alteração dos parâmetros do algoritmo para permitir a solução do planejamento de movimento e, também, devido à diferença na natureza dos obstáculos.

O resultado deste teste está ilustrado na Figura 36. O algoritmo foi capaz de solucionar o problema, porém foi necessário aumentar a abertura do corredor para 1 m, enquanto a abertura do corredor no Ambiente 2 é de 0,5 m. O espaço de busca foi aumentado para -5 m a 5 m na direção x e de -6 a 4 m na direção y. Além disso, foi

necessário reduzir a polarização da árvore para que, com uma maior exploração do ambiente, fosse possível sair do obstáculo na direção oposta ao destino. O fator de polarização ϵ da RRT foi reduzido de $1 \cdot 10^{-5}$ para $5 \cdot 10^{-6}$.



(a) (b)

Figura 36 – Trajetória gerada pela RRT proposta no ambiente com obstáculo não convexo: (a) projeção da RRT no espaço tridimensional e (b) movimentação resultante.

Vê-se, na Figura 35a, que houve ramos colidindo dentro do obstáculo em diversas direções, devido à baixa probabilidade de selecionar o destino por causa da redução de ϵ. Após a transposição do obstáculo, a RRT se expandiu em diversos ramos, aumentando a probabilidade de seleção do destino com o aumento de nós da árvore, até encontrar a posição final desejada.

O resultado deste teste mostra que o algoritmo proposto é promissor por ter sido capaz de encontrar uma trajetória em um ambiente com obstáculos não convexos, tarefa que os algoritmos baseados na minimização de função de custo encontrados na literatura não são capazes de solucionar devido à ocorrência de mínimos locais. Além disso, é de se esperar que melhorias no algoritmo resultem em um aumento na taxa de sucesso não só em ambientes com obstáculos convexos mas também não convexos, pois não houve necessidade de modificação do algoritmo. As alterações foram feitas apenas em seus parâmetros, como fator de polarização e espaço de busca.

5.7 Desafios observados com os experimentos

Os maiores problemas observados na técnica proposta foram a lentidão do algoritmo e a baixa taxa de sucesso nos ambientes testados, principalmente, no problema da janela curta.

No caso da lentidão do algoritmo, diversos fatores influenciam. Um deles é a dificuldade que o algoritmo RRT possui em atravessar passagens estreitas, que pode necessitar de um número alto de amostras até a transposição do caminho. Outro fator

ocorre durante a estabilização das oscilações da carga ao se aproximar do destino. Toda vez que uma amostra aleatória é sorteada ao invés do destino, a movimentação do quadricóptero causa novas oscilações que devem ser estabilizadas. O pior caso deste acontecimento ocorreu no Ambiente 1, que demorou cerca de 640 minutos para encontrar uma trajetória. No momento em que o sistema já estava estabilizando na posição de destino, um sorteio aleatório causou um giro agressivo no quadricóptero que foi necessária uma queda de cerca de 5 metros para estabilizar novamente o sistema. Com essa queda, o algoritmo teve que continuar explorando o ambiente até atingir novamente o destino.

Para solucionar o problema de lentidão, é possível otimizar diversas etapas do algoritmo e implementá-lo em uma linguagem de programação compilada, que deve reduzir bastante o tempo de execução da RRT. Além disso, abordagens mais robustas da RRT podem resolver o problema de transpor passagens estreitas, como os algoritmo propostos por Kamarry et al. (2015) e Santana et al. (2018) que fornecem modificações da RRT para facilitar a transposição em passagens estreitas e podem ser utilizados como base para adaptar o algoritmo proposto nesta dissertação. Adicionalmente, uma implementação adaptativa dos ganhos do controlador pode resultar em uma estabilização mais rápida do algoritmo quando estiver próximo do destino.

Com relação à baixa taxa de sucesso, conforme discutido na Seção 4.6, em que se apresenta a escolha do vizinho mais próximo, o algoritmos baseados na RRT clássica são bastante sensíveis à escolha da métrica de distância. Além disso, também é discutido o fato de que a distância Euclidiana não representa bem sistemas dinâmicos. Tais fatos são as causas da dificuldade de expansão da RRT observada nos experimentos realizados.

Tais problemas causam a escolha de nós que são selecionados como vizinhos mais próximos por diversas vezes. Quando um vizinho mais próximo é escolhido e leva à colisão, este mesmo nó tem altas chances de ser escolhido novamente até que um novo nó mais próximo da solução surja sem colidir com os obstáculos. Outra possibilidade que afeta o crescimento da RRT ocorre quando apenas um pequeno conjunto de nós são escolhidos como vizinhos mais próximos, levando à expansão de poucos ramos da RRT. Estes fatos ocorrem principalmente devido à polarização da árvore em que o destino final é amostrado por diversas vezes para acelerar a obtenção da solução.

Estes problemas podem ser solucionados de duas formas: selecionar uma nova métrica, o que pode ser inviável para sistemas não lineares, ou reduzir a sensibilidade do algoritmo à métrica escolhida. Esta segunda abordagem é bastante interessante e é discutida por Cheng e LaValle (2001).

Com o intuito de reduzir a sensibilidade à métrica de distância nesta dissertação, foram propostas duas regras de exclusão de nós na RRT, que são apresentadas na Seção 4.7. Então, foram verificadas as taxas de sucesso da RRT proposta com e sem exclusão de nós. Os resultados discutidos nesta seção estão resumidos na Tabela 10.

Tabela 10 – Comparação das taxas de sucesso com e sem exclusão de nós.

Característica	**Ambiente 2**	**Ambiente 3**	**Ambiente 4**
Com exclusão de nós	27%	43%	6%
Sem exclusão de nós	14%	37%	5%

É possível ver que houve um aumento considerável na taxa de sucesso, chegando praticamente dobrar o número de soluções encontradas para o Ambiente 2. Apesar disso, os resultados apresentados por Cheng e LaValle (2001) ainda são mais promissores, chegando a aumentar as taxas de sucesso de 8% para 98%. Acredita-se que este sucesso se deve à retropropagação da penalização de colisão para todos os nós parentes sempre que há uma colisão. Outro motivo da eficiência da abordagem está no momento de seleção do vizinho mais próximo, em que existe uma probabilidade, dependente da penalização de colisão, do nó ser ignorado e o próximo vizinho ser avaliado.

Como esta estratégia afeta diretamente as chances de um nó ser selecionado como vizinho mais próximo, a sensibilidade à métrica pode ser reduzida. Esta técnica não foi utilizada nesta dissertação porque a RRT implementada por Cheng e LaValle (2001) utiliza um espaço de ação discretizado que é utilizado para estimar a probabilidade de colisão. No caso da RRT proposta nesta dissertação, a ação é um resultado do controlador. Portanto, deve-se fazer uma adaptação para que uma abordagem semelhante possa ser efetiva.

É importante, também, citar que quanto maior o número máximo de nós para resolver o problema, maior será a taxa de sucesso do algoritmo.Em um teste realizado por LaValle et al. (2001), um sistema com um espaço de estados de 12 dimensões é testado e foram necessários 23800 nós para resolver o problema. Este fato pode indicar que para resolver com eficiência o problema estudado nesta dissertação, que possui um espaço de estados de 24 dimensões, possam ser necessários muito mais que 25000 nós.

6 Conclusões

O foco deste trabalho foi o estudo, desenvolvimento e a implementação de um planejador de movimentos para um sistema de transporte de cargas suspensas por um quadricóptero. Inicialmente, foi feita uma revisão bibliográfica a respeito do tema de planejadores de movimento e controle aplicados em sistemas de transporte de carga com quadricópteros e, em seguida, foi realizada uma revisão bibliográfica focada no planejamento de movimento baseados em Árvores Aleatórias de Exploração Rápida (RRT) considerando restrições dinâmicas e cinemáticas.

Durante a revisão bibliográfica, foram encontradas várias soluções da literatura para o controle do sistema de transporte de carga que eram focadas apenas na supressão das oscilações da carga. Apenas algumas abordagens mais recentes focaram no desenvolvimento de controladores capazes de rastrear a posição da carga de acordo com uma trajetória desejada, podendo conter ou não oscilações. Com controladores mais robustos, os artigos focaram no planejamento de movimento. É comum que estes algoritmos considerem as restrições do sistema, que podem ser cinemáticas, holonômicas e até dinâmicas.

Também durante a etapa de revisão bibliográfica, foi verificado que vários trabalhos que propuseram planejadores de movimentos para o transporte de carga utilizando quadricópteros focavam na minimização de funções de custo. Os resultados mostraram a eficiência dos planejadores de movimento em encontrar caminhos factíveis de tarefas complexas. Entre as tarefas, a mais testada era fazer com que o quadricóptero atravessasse, junto com a carga, uma janela cujo tamanho é menor que o comprimento do sistema. Esta é uma tarefa complexa e interessante de investigar, pois a solução do problema envolve encontrar manobras livres de colisão que são complexas de realizar devido à natureza não linear e sub-atuada do problema.

Uma desvantagem encontrada nas abordagens da literatura foi a simplificação dos obstáculos para facilitar a convergência das técnicas de otimização. Considerando esta desvantagem, foi proposta uma abordagem diferente que resolvesse esse problema. Dentro das técnicas de planejamento de movimento estudadas, a mais adequada para o problema foi a RRT, pois não depende de otimização de funções, não requer restrições de obstáculos e é eficiente em problemas de alta dimensionalidade. Em contrapartida, a RRT obtém trajetórias sub-ótimas e há a possibilidade de não se obter uma solução dentro do limite de nós, caso ela exista.

Para a implementação e proposição da RRT adequada para o problema da transposição de obstáculos que requerem movimentação agressiva, foram estudados temas relacionados à modelagem de sistemas dinâmicos, geometria diferencial, controle não linear

e planejadores de movimento. Então, o sistema, que consiste em um quadricóptero com carga suspensa via cabo, foi modelado e estabilizado utilizando um controlador geométrico não linear. Em seguida, foi proposta uma RRT baseada no controlador geométrico.

Foram feitos experimentos simulados para avaliar o sistema em quatro tipos de ambientes e, entre eles, o problema da janela curta, que envolve o planejamento de movimento através de uma janela cuja altura é cerca de 85% do comprimento do quadricóptero com a carga. A partir dos experimentos simulados, foi verificado que a RRT proposta foi capaz de solucionar todos os ambientes propostos. Entretanto, houve uma baixa taxa de sucesso nos ambientes experimentados.

O algoritmo proposto é uma primeira investigação na área de planejamento de movimento para o transporte de carga suspensa com um quadricóptero utilizando movimentações agressivas. Apesar do desempenho inferior a algoritmos já apresentados na literatura, esta abordagem se mantém relevante na área de planejamento de movimento. Principalmente, porque o algoritmo também se mostrou capaz em resolver o planejamento em um ambiente com obstáculos não convexos, enquanto as abordagens da literatura são limitadas à solução de ambientes com obstáculos convexos.

As investigações feitas para avaliar as causas da baixa taxa de sucesso indicam que fatores como a dificuldade da RRT em atravessar obstáculos apertados e em se expandir pelo ambiente como consequência da sensibilidade do algoritmo à métrica Euclidiana reduziram a capacidade de expansão de ramos na RRT. Com relação à sensibilidade da RRT à métrica escolhida, entendeu-se que a penalização de nós que colidem ajudam a melhorar a eficiência do algoritmo. Este fato foi corroborado com experimentos que mostraram que as regras de exclusão de nós ajudaram a melhorar a taxa de sucesso do algoritmo.

Para trabalhos futuros, é possível desenvolver uma estratégia de penalização de colisão mais eficiente com o intuito de aumentar a taxa de sucesso do algoritmo. Outro aspecto que pode ser melhorado é o tempo de execução do algoritmo para viabilizar sua utilização em tempo real. Adicionalmente, experimentos em novos ambientes com obstáculos não convexos podem ser realizados, principalmente para comparar os efeitos da melhoria do algoritmo na taxa de sucesso para os dois tipos de ambientes, com obstáculos convexos e não convexos. É também importante verificar se a RRT ainda se mantém probabilisticamente completa após as modificações propostas.

Além disso, percebe-se a necessidade de realização de experimentos com quadricópteros reais para verificar de fato a exequibilidade das soluções obtidas. Com experimentos reais, também é importante investigar os efeitos aerodinâmicos dos motores que podem causar turbulência na carga.

Referências

AMUKELE, T. et al. Drone transportation of blood products. *Transfusion*, Wiley Online Library, v. 57, n. 3, p. 582–588, 2017. Citado na página 1.

ASCTEC. *ASCTEC Hummingbird: Safety data sheet*. 2015. Http://www.asctec.de/downloads/public/HB_AscTec-Hummingbird_safety-data-sheet.pdf. Citado na página 30.

BASCETTA, L.; ARRIETA, I. M.; PRANDINI, M. Flat-rrt*: A sampling-based optimal trajectory planner for differentially flat vehicles with constrained dynamics. *IFAC-PapersOnLine*, Elsevier, v. 50, n. 1, p. 6965–6970, 2017. Citado na página 10.

BULLO, F.; LEWIS, A. D. *Geometric Control of Mechanical Systems*. New York-Heidelberg-Berlin: Springer Verlag, 2004. (Texts in Applied Mathematics, v. 49). ISBN 0-387-22195-6. Citado 5 vezes nas páginas 20, 75, 78, 79 e 80.

CANO, R. et al. Mechanical design of a 6-dof aerial manipulator for assembling bar structures using uavs. In: *2nd RED-UAS 2013 workshop on research, education and development of unmanned aerial systems*. [S.l.: s.n.], 2013. v. 218. Citado na página 1.

CHENG, P.; LAVALLE, S. M. Reducing metric sensitivity in randomized trajectory design. In: IEEE. *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*. [S.l.], 2001. v. 1, p. 43–48. Citado 5 vezes nas páginas 37, 38, 39, 60 e 61.

CROUSAZ, C. D.; FARSHIDIAN, F.; BUCHLI, J. Aggressive optimal control for agile flight with a slung load. In: CITESEER. *IROS 2014 Workshop on Machine Learning in Planning and Control of Robot Motion*. [S.l.], 2014. Citado 4 vezes nas páginas 2, 6, 8 e 34.

CRUZ, P.; FIERRO, R. Autonomous lift of a cable-suspended load by an unmanned aerial robot. In: IEEE. *2014 IEEE Conference on Control Applications (CCA)*. [S.l.], 2014. p. 802–807. Citado na página 8.

FARRELL, J. A. et al. Command filtered backstepping. *IEEE Transactions on Automatic Control*, IEEE, v. 54, n. 6, p. 1391–1395, 2009. Citado na página 22.

FAUST, A. et al. Learning swing-free trajectories for uavs with a suspended load. In: IEEE. *2013 IEEE International Conference on Robotics and Automation*. [S.l.], 2013. p. 4902–4909. Citado 2 vezes nas páginas 2 e 5.

FENG, Y. et al. Adaptive controller design for generic quadrotor aircraft platform subject to slung load. In: IEEE. *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*. [S.l.], 2015. p. 1135–1139. Citado na página 6.

GEISERT, M.; MANSARD, N. Trajectory generation for quadrotor based systems using numerical optimal control. In: IEEE. *2016 IEEE international conference on robotics and automation (ICRA)*. [S.l.], 2016. p. 2958–2964. Citado 4 vezes nas páginas 3, 7, 8 e 34.

GLASSMAN, E.; TEDRAKE, R. A quadratic regulator-based heuristic for rapidly exploring state space. In: IEEE. *2010 IEEE International Conference on Robotics and Automation*. [S.l.], 2010. p. 5021–5028. Citado na página 38.

GOLDSTEIN, H.; POOLE, C.; SAFKO, J. *Classical mechanics*. [S.l.]: AAPT, 2002. Citado 2 vezes nas páginas 15 e 16.

GOODARZI, F. A.; LEE, D.; LEE, T. Geometric stabilization of a quadrotor uav with a payload connected by flexible cable. In: IEEE. *2014 American Control Conference*. [S.l.], 2014. p. 4925–4930. Citado na página 8.

HARDY, A. et al. Using low-cost drones to map malaria vector habitats. *Parasites & vectors*, BioMed Central, v. 10, n. 1, p. 29, 2017. Citado na página 1.

HODGSON, A.; PEEL, D.; KELLY, N. Unmanned aerial vehicles for surveying marine fauna: assessing detection probability. *Ecological Applications*, Wiley Online Library, v. 27, n. 4, p. 1253–1267, 2017. Citado na página 1.

KAMARRY, S. et al. Compact rrt: A new approach for guided sampling applied to environment representation and path planning in mobile robotics. In: IEEE. *2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)*. [S.l.], 2015. p. 259–264. Citado na página 60.

KARAMAN, S.; FRAZZOLI, E. Sampling-based algorithms for optimal motion planning. *The international journal of robotics research*, Sage Publications Sage UK: London, England, v. 30, n. 7, p. 846–894, 2011. Citado na página 9.

KARAMAN, S. et al. Anytime motion planning using the rrt. In: IEEE. *2011 IEEE International Conference on Robotics and Automation*. [S.l.], 2011. p. 1478–1483. Citado 2 vezes nas páginas 9 e 24.

KUFFNER, J. J.; LAVALLE, S. M. Rrt-connect: An efficient approach to single-query path planning. In: IEEE. *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. [S.l.], 2000. v. 2, p. 995–1001. Citado na página 24.

KUI, Y. et al. Sliding mode control for a quadrotor slung load system. In: IEEE. *Control Conference (CCC), 2017 36th Chinese*. [S.l.], 2017. p. 3697–3703. Citado na página 6.

KUMAR, V.; MICHAEL, N. Opportunities and challenges with autonomous micro aerial vehicles. *The International Journal of Robotics Research*, SAGE Publications Sage UK: London, England, v. 31, n. 11, p. 1279–1291, 2012. Citado na página 1.

LAVALLE, S. M. Rapidly-exploring random trees: A new tool for path planning. Citeseer, 1998. Citado 2 vezes nas páginas 8 e 35.

LAVALLE, S. M.; JR, J. J. K. Rapidly-exploring random trees: Progress and prospects. Citeseer, 2000. Citado na página 23.

LAVALLE, S. M.; JR, J. J. K. Randomized kinodynamic planning. *The international journal of robotics research*, SAGE Publications, v. 20, n. 5, p. 378–400, 2001. Citado 2 vezes nas páginas 26 e 38.

LAVALLE, S. M. et al. Rapidly-exploring random trees: Progress and prospects. *Algorithmic and computational robotics: new directions*, n. 5, p. 293–308, 2001. Citado 2 vezes nas páginas 23 e 61.

LEE, T. *Computational Geometric Mechanics and Control of Rigid Bodies.* Tese (Doutorado), 2008. Citado na página 15.

LEE, T.; LEOK, M.; MCCLAMROCH, N. H. Lagrangian mechanics and variational integrators on two-spheres. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 79, n. 9, p. 1147–1174, 2009. Citado na página 15.

LEE, T.; LEOK, M.; MCCLAMROCH, N. H. Control of complex maneuvers for a quadrotor uav using geometric methods on se (3). *arXiv preprint arXiv:1003.2005*, 2010. Citado 3 vezes nas páginas 16, 17 e 18.

LEE, T.; MCCLAMROCH, N. H.; LEOK, M. A lie group variational integrator for the attitude dynamics of a rigid body with applications to the 3d pendulum. In: IEEE. *Proceedings of 2005 IEEE Conference on Control Applications, 2005. CCA 2005.* [S.l.], 2005. p. 962–967. Citado na página 15.

LUCASSEN, L. R.; STERK, F. J. Dynamic stability analysis of a hovering helicopter with a sling load. *Journal of the American Helicopter Society*, Vertical Flight Society, v. 10, n. 2, p. 6–12, 1965. Citado na página 2.

MAZUR, M.; WISNIEWSKI, A.; MCMILLAN, J. Clarity from above: Pwc global report on the commercial applications of drone technology. *Warsaw: Drone Powered Solutions, PriceWater house Coopers*, 2016. Citado na página 1.

MELLINGER, D. et al. Design, modeling, estimation and control for aerial grasping and manipulation. In: IEEE. *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.], 2011. p. 2668–2673. Citado na página 1.

MICHAEL, N. et al. The grasp multiple micro-uav testbed. *IEEE Robotics & Automation Magazine*, IEEE, v. 17, n. 3, p. 56–65, 2010. Citado na página 30.

MO, R.; GENG, Q.; LU, X. Study on control method of a rotor uav transportation with slung-load. In: IEEE. *Control Conference (CCC), 2016 35th Chinese*. [S.l.], 2016. p. 3274–3279. Citado na página 6.

MOE, M. et al. *Eye in the sky. Available via Statista, the statistics portal*. 2017. Disponível em: <www.a2apple.com/eye-in-the-sky/>. Acesso em: 01 dec. 2019. Citado na página 1.

MOLINA, M. d. R. de M.; CAMPOS, V. S. *Ethics and Civil Drones: European Policies and Proposals for the Industry*. [S.l.]: Springer Open, 2018. Citado na página 1.

PALUNKO, I.; CRUZ, P.; FIERRO, R. Agile load transportation: Safe and efficient load manipulation with aerial robots. *IEEE robotics & automation magazine*, IEEE, v. 19, n. 3, p. 69–79, 2012. Citado na página 5.

PALUNKO, I. et al. A reinforcement learning approach towards autonomous suspended load manipulation using aerial robots. In: IEEE. *2013 IEEE International Conference on Robotics and Automation*. [S.l.], 2013. p. 4896–4901. Citado na página 5.

PALUNKO, I.; FIERRO, R.; CRUZ, P. Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: A dynamic programming approach. In: IEEE. *2012 IEEE International Conference on Robotics and Automation*. [S.l.], 2012. p. 2691–2697. Citado na página 2.

PANDIT, V.; POOJARI, A. A study on amazon prime air for feasibility and profitability: A graphical data analysis. *IOSR Journal of Business and Management*, v. 16, n. 11, p. 06–11, 2014. Citado na página 1.

PEREZ, A. et al. Lqr-rrt*: Optimal sampling-based motion planning with automatically derived extension heuristics. In: IEEE. *2012 IEEE International Conference on Robotics and Automation*. [S.l.], 2012. p. 2537–2542. Citado 5 vezes nas páginas 9, 23, 24, 34 e 37.

POWERS, C.; MELLINGER, D.; KUMAR, V. Quadrotor kinematics and dynamics. *Handbook of Unmanned Aerial Vehicles*, Springer, p. 307–328, 2015. Citado na página 30.

SADR, S.; MOOSAVIAN, S. A. A.; ZARAFSHAN, P. Dynamics modeling and control of a quadrotor with swing load. *Journal of Robotics*, Hindawi, v. 2014, 2014. Citado na página 5.

SANKEY, T. et al. Uav lidar and hyperspectral fusion for forest monitoring in the southwestern usa. *Remote Sensing of Environment*, Elsevier, v. 195, p. 30–43, 2017. Citado na página 1.

SANTANA, A. et al. The polarized rrt-edge approach. In: IEEE. *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*. [S.l.], 2018. p. 271–276. Citado na página 60.

SHKOLNIK, A. et al. Bounding on rough terrain with the littledog robot. *The International Journal of Robotics Research*, SAGE Publications Sage UK: London, England, v. 30, n. 2, p. 192–215, 2011. Citado na página 9.

SHKOLNIK, A.; TEDRAKE, R. Path planning in 1000+ dimensions using a task-space voronoi bias. In: IEEE. *2009 IEEE International Conference on Robotics and Automation*. [S.l.], 2009. p. 2061–2067. Citado 3 vezes nas páginas 9, 23 e 26.

SHKOLNIK, A.; WALTER, M.; TEDRAKE, R. Reachability-guided sampling for planning under differential constraints. In: IEEE. *2009 IEEE International Conference on Robotics and Automation*. [S.l.], 2009. p. 2859–2865. Citado 3 vezes nas páginas VII, 24 e 26.

SPONG, M. W.; HUTCHINSON, S.; VIDYASAGAR, M. Robot modeling and control, jon wiley & sons. *Inc., ISBN-100-471-649*, 2005. Citado na página 11.

SREENATH, K.; LEE, T.; KUMAR, V. Geometric control and differential flatness of a quadrotor uav with a cable-suspended load. In: IEEE. *52nd IEEE Conference on Decision and Control*. [S.l.], 2013. p. 2269–2274. Citado 11 vezes nas páginas 2, 7, 8, 15, 16, 17, 20, 21, 22, 30 e 33.

SREENATH, K.; MICHAEL, N.; KUMAR, V. Trajectory generation and control of a quadrotor with a cable-suspended load-a differentially-flat hybrid system. In: IEEE. *2013 IEEE International Conference on Robotics and Automation*. [S.l.], 2013. p. 4888–4895. Citado na página 29.

SUN, B. et al. Trajectory planning of quadrotor uav with suspended payload based on predictive control. In: IEEE. *2018 37th Chinese Control Conference (CCC)*. [S.l.], 2018. p. 10049–10054. Citado na página 6.

TANG, S.; KUMAR, V. Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload. In: IEEE. *2015 IEEE International Conference on Robotics and Automation (ICRA)*. [S.l.], 2015. p. 2216–2222. Citado 3 vezes nas páginas 3, 7 e 34.

TANG, S.; SREENATH, K.; KUMAR, V. Aggressive maneuvering of a quadrotor with a cable-suspended payload. In: *Robotics: Science and Systems, Workshop on Women in Robotics*. [S.l.: s.n.], 2014. Citado 2 vezes nas páginas 2 e 15.

TANG, S.; WÜEST, V.; KUMAR, V. Aggressive flight with suspended payloads using vision-based control. *IEEE Robotics and Automation Letters*, IEEE, v. 3, n. 2, p. 1152–1159, 2018. Citado 3 vezes nas páginas 7, 8 e 32.

WEBB, D. J.; BERG, J. V. D. Kinodynamic rrt*: Asymptotically optimal motion planning for robots with linear dynamics. In: IEEE. *2013 IEEE International Conference on Robotics and Automation*. [S.l.], 2013. p. 5054–5061. Citado na página 10.

ZENG, J.; KOTARU, P.; SREENATH, K. Geometric control and differential flatness of a quadrotor uav with load suspended from a pulley. In: IEEE. *2019 American Control Conference (ACC)*. [S.l.], 2019. p. 2420–2427. Citado na página 8.

Apêndices

APÊNDICE A – Fundamentos matemáticos sobre os *manifolds* S^2 e $SO(3)$

Com as definições apresentadas na Seção 3.1, a orientação do quadricóptero pode ser expressada por uma matriz de rotação $\boldsymbol{R}$ que pertence ao Grupo Especial Ortogonal $SO(3)$ e descreve a orientação relativa entre dois sistemas de coordenadas. O *manifold* $SO(3)$ é definido por

$$SO(3) = \{\boldsymbol{R}^{3\times 3} \mid \boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{I}, \det[\boldsymbol{R}] = +1\}. \tag{A.1}$$

Uma representação abstrata do $SO(3)$ juntamente com seu espaço tangente na matriz identidade está apresentada na Figura 1a. Como o espaço tangente representa o conjunto de possíveis derivadas direcionais em um ponto, é possível obter sua relação matemática derivando a matriz de rotação.

Seja uma matriz de rotação variando no tempo de forma que $\boldsymbol{R}(t) \in SO(3)\ \forall\ t \in \mathbb{R}$. Assumindo que $\boldsymbol{R}(t)$ é continuamente diferenciável com relação a t, tem-se que

$$\boldsymbol{R}(t)^T\boldsymbol{R}(t) = \boldsymbol{I}. \tag{A.2}$$

Diferenciando ambos os lados com relação a t

$$\dot{\boldsymbol{R}}(t)^T\boldsymbol{R}(t) + \boldsymbol{R}(t)^T\dot{\boldsymbol{R}}(t) = \boldsymbol{0}, \tag{A.3}$$

fazendo $\hat{\boldsymbol{\Omega}} = \boldsymbol{R}(t)^T\dot{\boldsymbol{R}}(t)$ e substituindo em (A.3) tem-se que

$$\hat{\boldsymbol{\Omega}} = -\hat{\boldsymbol{\Omega}}^T. \tag{A.4}$$

Com isso, pode-se concluir que $\hat{\boldsymbol{\Omega}}$ é uma matriz anti-simétrica cujo grupo é definido por

$$\mathfrak{so}(3) = \{\boldsymbol{S} \in \mathbb{R}^{3x3} \mid \boldsymbol{S} = -\boldsymbol{S}^T\}, \tag{A.5}$$

que representa o espaço tangente na identidade da figura 1a.

O símbolo $\hat{(.)}$ em $\hat{\boldsymbol{\Omega}}$ caracteriza um isomorfismo entre $\mathbb{R}^3$ e $\mathfrak{so}(3)$ e representa um produto vetorial como uma transformação linear, ou seja, $\hat{\boldsymbol{x}}\boldsymbol{y} = \boldsymbol{x} \times \boldsymbol{y}$ para quaisquer $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^3$. Sua operação inversa é representada por $(.)^\vee$. O vetor $\boldsymbol{\Omega} \in \mathbb{R}^3$ pode ser entendido como um vetor velocidade angular que age no eixo sobre o qual ocorre uma rotação. O mapeamento entre o vetor velocidade angular $\boldsymbol{\Omega} \in \mathbb{R}^3$ e $\hat{\boldsymbol{\Omega}} \in \mathfrak{so}(3)$ pode ser escrito da

seguinte forma

$$\hat{\boldsymbol{\Omega}} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}, \tag{A.6}$$

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix}^{\vee} = \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix}. \tag{A.7}$$

Uma propriedade desta transformação que é utilizada neste trabalho, resultante da propriedade de produto vetorial, é

$$\hat{\boldsymbol{x}}\boldsymbol{y} = \boldsymbol{x} \times \boldsymbol{y} = -\boldsymbol{y} \times \boldsymbol{x} = -\hat{\boldsymbol{y}}\boldsymbol{x}. \tag{A.8}$$

Com o vetor velocidade angular, é possível obter a matriz de rotação a partir do mapeamento exponencial $exp : \mathfrak{so}(3) \to SO(3)$

$$\boldsymbol{R} = exp(\hat{\boldsymbol{\Omega}}). \tag{A.9}$$

O mapeamento exponencial é essencial, pois é possível obter variações de matrizes de rotação a partir de uma soma vetorial no espaço tangente, como pode ser visto na Figura 1a.

Com relação ao pêndulo esférico, a orientação da massa pontual pode ser obtida a partir de um vetor unitário $\boldsymbol{p}$ que aponta da base do pêndulo até a massa pontual. Na Figura 2, a base do pêndulo é a origem do sistema de coordenadas fixo ao quadricóptero e o vetor unitário aponta para a carga.

Devido à restrição de comprimento unitário, o vetor $\boldsymbol{p}$ situa-se na casca de uma esfera unitária. Este espaço é chamado de 2-esfera e corresponde ao o conjunto de pontos que possuem comprimento unitário para a origem no $\mathbb{R}^3$, ou seja,

$$S^2 = \{\boldsymbol{p} \in \mathbb{R}^3 \mid \boldsymbol{p} \cdot \boldsymbol{p} = 1\}. \tag{A.10}$$

O espaço tangente $\mathrm{T}_{\boldsymbol{p}}S^2$ em $\boldsymbol{p} \in S^2$ é o plano tangente à 2-esfera no ponto $\boldsymbol{p}$. O espaço 2-esfera e seu espaço tangente no ponto $\boldsymbol{p}$ estão ilustrados na Figura 1b. Como todas as possíveis derivadas de um ponto situam-se em seu espaço tangente, uma curva $\boldsymbol{p} : \mathbb{R} \to S^2$ e sua derivada no tempo $\dot{\boldsymbol{p}}$ são vetores tangentes, portanto, devem satisfazer

$$\boldsymbol{p} \cdot \dot{\boldsymbol{p}} = 0. \tag{A.11}$$

A derivada no tempo de $\boldsymbol{p}$ também pode ser escrita da seguinte forma

$$\dot{\boldsymbol{p}} = \boldsymbol{\omega} \times \boldsymbol{p}, \tag{A.12}$$

em que $\boldsymbol{\omega} \in \mathbb{R}^3$ é o vetor velocidade angular restrito a ser ortogonal a $\boldsymbol{p}$, portanto

$$\boldsymbol{p} \cdot \boldsymbol{\omega} = 0, \tag{A.13}$$

derivando (A.13) segue também que

$$\boldsymbol{p} \cdot \dot{\boldsymbol{\omega}} = 0. \tag{A.14}$$

As relações de (A.11) a (A.14) são utilizadas para o desenvolvimento das equações de movimento do sistema quadricóptero-carga.

APÊNDICE B – Obtenção das equações de movimento

Para aplicar (3.6) ao sistema quadricóptero-carga, é necessário definir o conjunto de coordenadas generalizadas, que são a menor quantidade de coordenadas capazes de representar o sistema por completo. Para o caso do sistema em questão, as coordenadas generalizadas são $\boldsymbol{q} = \{\boldsymbol{x}_L, \boldsymbol{p}, \boldsymbol{R}\}$. Então, o espaço de configuração desse sistema é $\mathbb{R}^3 \times \mathbb{S}^2 \times SO(3)$ e seu estado é dado por

$$x = [\boldsymbol{x}_L, \dot{\boldsymbol{x}}_L, \boldsymbol{p}, \boldsymbol{\omega}, \boldsymbol{R}, \boldsymbol{\Omega}]^T. \tag{B.1}$$

Deste modo, é possível obter uma expressão para o Lagrangiano começando com a energia cinética do sistema $\mathcal{T}$. A energia cinética total do sistema é dada pelo somatório da energia translacional do quadricóptero e da carga, bem como da energia rotacional do quadricóptero, ou seja,

$$\mathcal{T} = \frac{1}{2} m_Q \dot{\boldsymbol{x}}_Q \cdot \dot{\boldsymbol{x}}_Q + \frac{1}{2} m_L \dot{\boldsymbol{x}}_L \cdot \dot{\boldsymbol{x}}_L + \frac{1}{2} \boldsymbol{\Omega} \cdot \boldsymbol{J}_Q \cdot \boldsymbol{\Omega}, \tag{B.2}$$

em que o vetor velocidade angular $\boldsymbol{\Omega}$ está representado com relação ao sistema de coordenadas fixo ao corpo do quadricóptero e a posição do quadricóptero está relacionada com a posição da carga a partir das seguintes restrições

$$\boldsymbol{x}_Q = \boldsymbol{x}_L - l\boldsymbol{p}, \tag{B.3}$$

$$\dot{\boldsymbol{x}}_Q = \dot{\boldsymbol{x}}_L - l\dot{\boldsymbol{p}}. \tag{B.4}$$

Ao substituir (B.4) em (B.2), obtém-se

$$\begin{aligned} \mathcal{T} &= \frac{1}{2} m_Q (\dot{\boldsymbol{x}}_L - l\dot{\boldsymbol{p}}) \cdot (\dot{\boldsymbol{x}}_L - l\dot{\boldsymbol{p}}) + \frac{1}{2} m_L \dot{\boldsymbol{x}}_L \cdot \dot{\boldsymbol{x}}_L + \frac{1}{2} \boldsymbol{\Omega} \cdot \boldsymbol{J}_Q \cdot \boldsymbol{\Omega} \\ &= \frac{1}{2}(m_Q + m_L)\dot{\boldsymbol{x}}_L \cdot \dot{\boldsymbol{x}}_L - m_Q l \dot{\boldsymbol{x}}_L \cdot \dot{\boldsymbol{p}} + \frac{1}{2} m_Q l^2 \dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}} + \frac{1}{2} \boldsymbol{\Omega} \cdot \boldsymbol{J}_Q \cdot \boldsymbol{\Omega}. \end{aligned} \tag{B.5}$$

A energia potencial $\mathcal{U}$ é obtida a partir dos termos de energia potencial gravitacional da carga e do quadricóptero. Ao utilizar a restrição (B.3) para substituir a posição do quadricóptero, a energia potencial é dada por

$$\begin{aligned} \mathcal{U} &= m_Q g \boldsymbol{x}_Q \cdot \boldsymbol{e}_3 + m_L g \boldsymbol{x}_L \cdot \boldsymbol{e}_3 \\ &= m_Q g (\boldsymbol{x}_L - l\boldsymbol{p}) \cdot \boldsymbol{e}_3 + m_L g \boldsymbol{x}_L \cdot \boldsymbol{e}_3 \\ &= (m_Q + m_L) g \boldsymbol{x}_L \cdot \boldsymbol{e}_3 - m_Q g l \boldsymbol{p} \cdot \boldsymbol{e}_3. \end{aligned} \tag{B.6}$$

Após obter os termos necessários para encontrar uma expressão para $\mathcal{L}$, é preciso encontrar um modo de se obter sua variação $\delta\mathcal{L}$. Para tal, utiliza-se a aproximação de primeira ordem da série de Taylor. Então, a variação da energia cinética é dada por

$$\begin{aligned}\delta\mathcal{T} &= \mathcal{T}(\dot{\boldsymbol{x}}_L + \delta\dot{\boldsymbol{x}}_L, \dot{\boldsymbol{p}} + \delta\dot{\boldsymbol{p}}, \boldsymbol{\Omega} + \delta\boldsymbol{\Omega}) - \mathcal{T}(\dot{\boldsymbol{x}}_L, \dot{\boldsymbol{p}}, \boldsymbol{\Omega}),\\ &\approx \frac{\partial\mathcal{T}}{\partial\dot{\boldsymbol{x}}_L}\delta\dot{\boldsymbol{x}}_L + \frac{\partial\mathcal{T}}{\partial\dot{\boldsymbol{p}}}\delta\dot{\boldsymbol{p}} + \frac{\partial\mathcal{T}}{\partial\boldsymbol{\Omega}}\delta\boldsymbol{\Omega},\\ &\approx ((m_Q + m_L)\dot{\boldsymbol{x}}_L - m_Q l\dot{\boldsymbol{p}})\delta\dot{\boldsymbol{x}}_L + (-m_Q l\dot{\boldsymbol{x}}_L + m_Q l^2\dot{\boldsymbol{p}})\delta\dot{\boldsymbol{p}} + \boldsymbol{J}_Q\boldsymbol{\Omega}\cdot\delta\boldsymbol{\Omega}. \qquad \text{(B.7)}\end{aligned}$$

Para chegar nesta expressão, foi utilizado o fato de que a matriz de inércia é simétrica e que

$$\frac{1}{2}\frac{\partial(\boldsymbol{\Omega}\cdot\boldsymbol{J}_Q\cdot\boldsymbol{\Omega})}{\partial\boldsymbol{\Omega}} = \frac{1}{2}(\boldsymbol{J}_Q + \boldsymbol{J}_Q^T)\boldsymbol{\Omega} = \boldsymbol{J}_Q\boldsymbol{\Omega}.$$

A variação da energia potencial é obtida de maneira semelhante à da energia cinética

$$\begin{aligned}\delta\mathcal{U} &= \mathcal{U}(\boldsymbol{x}_L + \delta\boldsymbol{x}_L, \boldsymbol{p} + \delta\boldsymbol{p}) - \mathcal{U}(\boldsymbol{x}_L, \boldsymbol{p}),\\ \delta\mathcal{U} &\approx \frac{\partial\mathcal{U}}{\partial\boldsymbol{x}_L}\delta\boldsymbol{x}_L + \frac{\partial\mathcal{U}}{\partial\boldsymbol{p}}\delta\boldsymbol{p}\\ \delta\mathcal{U} &\approx (m_Q + m_L)g\boldsymbol{e}_3\cdot\delta\boldsymbol{x}_L - m_Q gl\boldsymbol{e}_3\cdot\delta\boldsymbol{p}. \qquad \text{(B.8)}\end{aligned}$$

Resta encontrar a variação referente às forças e momentos externos δW. Este termo que surge no Princípio de Hamilton é chamado de trabalho virtual e pode ser decomposto em

$$\delta W = \delta W_F + \delta W_M, \qquad \text{(B.9)}$$

em que δW_F é o trabalho virtual referente à força resultante e δW_M ao momento resultante.

Vale lembrar que as coordenadas generalizadas do sistema quadricóptero-carga são $\boldsymbol{q} = \{\boldsymbol{x}_L, \boldsymbol{p}, \boldsymbol{R}\}$. Então, o trabalho virtual da força é obtido ao calcular a contribuição da força resultante, que age no centro de massa, para cada coordenada generalizada da seguinte maneira

$$\begin{aligned}\delta W_F &= f\boldsymbol{R}\boldsymbol{e}_3\cdot\sum_{j=1}^{3}\frac{\partial\boldsymbol{x}_Q}{\partial\boldsymbol{q}_j}\delta\boldsymbol{q}_j\\ &= f\boldsymbol{R}\boldsymbol{e}_3\cdot\sum_{j=1}^{3}\frac{\partial(\boldsymbol{x}_L - l\boldsymbol{p})}{\partial\boldsymbol{q}_j}\delta\boldsymbol{q}_j\\ &= f\boldsymbol{R}\boldsymbol{e}_3\cdot(\delta\boldsymbol{x}_L - l\delta\boldsymbol{p}). \qquad \text{(B.10)}\end{aligned}$$

No caso do trabalho virtual referente ao momento gerado pelos motores, sua expressão pode ser obtida a partir da contribuição do momento que age na velocidade angular do corpo do quadricóptero para cada derivada da coordenada generalizada, ou seja,

$$\begin{aligned}\delta W_M &= \boldsymbol{M} \cdot \sum_{j=1}^{3} \frac{\partial \boldsymbol{\Omega}}{\partial \dot{\boldsymbol{q}}_j} \delta \boldsymbol{q}_j \\ &= \boldsymbol{M} \cdot \frac{\partial (\boldsymbol{R}^T \dot{\boldsymbol{R}})^\vee}{\partial \dot{\boldsymbol{R}}} \delta \boldsymbol{R} \\ &= \boldsymbol{M} \cdot (\boldsymbol{R}^T \delta \boldsymbol{R})^\vee. \end{aligned} \tag{B.11}$$

Então, a integral de ação (3.6) pode ser reescrita como

$$\delta S = \int_{t_1}^{t_2} \left(\delta W_F + \delta W_M + \delta \mathcal{T} - \delta \mathcal{U}\right)\, dt = 0. \tag{B.12}$$

Substituindo (B.7), (B.8), (B.10) e (B.11) em (B.12) tem-se

$$\begin{aligned}\delta S = \quad & \int_{t_1}^{t_2} \Big(\Big((m_Q + m_L)\dot{\boldsymbol{x}}_L - m_Q l \dot{\boldsymbol{p}} \Big) \cdot \delta \dot{\boldsymbol{x}}_L + \Big(f \boldsymbol{R} \boldsymbol{e}_3 - (m_Q + m_L) g \boldsymbol{e}_3 \Big) \cdot \delta \boldsymbol{x}_L \Big) dt \\ & + \int_{t_1}^{t_2} \Big((m_Q l^2 \dot{\boldsymbol{p}} - m_Q l \dot{\boldsymbol{x}}_L) \cdot \delta \dot{\boldsymbol{p}} + (m_Q g l \boldsymbol{e}_3 - f \boldsymbol{R} \boldsymbol{e}_3) \cdot \delta \boldsymbol{p} \Big) dt \\ & + \int_{t_1}^{t_2} \Big((\boldsymbol{J}_Q \boldsymbol{\Omega}) \cdot \delta \boldsymbol{\Omega} + \boldsymbol{M} \cdot (\boldsymbol{R}^T \delta \boldsymbol{R})^\vee \Big) dt = 0. \end{aligned} \tag{B.13}$$

Em (B.13), os termos $\boldsymbol{x}_L$ e $\dot{\boldsymbol{x}}_L$ variam no $\mathbb{R}^3$. Porém, os termos $\delta \boldsymbol{R}$ e $\delta \boldsymbol{p}$ variam em seus respectivos *manifolds* e não podem ser utilizados diretamente para encontrar as equações de movimento. Como um *manifold* é um espaço topológico que localmente assemelha-se a um espaço Euclidiano é possível encontrar variações infinitesimais, que se encontram no espaço tangente ao *manifold* e variam em um espaço Euclidiano.

Bullo e Lewis (2004) apresentam um método para encontrar as variações infinitesimais em um *manifold* Q. Para isso, é definido um mapeamento $\gamma(t) : [a, b] \to Q$ que representa uma trajetória começando no ponto a e terminando no ponto b. Em seguida, define-se um segundo mapeamento $\gamma^\epsilon(t)$ que possui uma trajetória ligeiramente diferente, parametrizada por ϵ, mas que começa e termina nos mesmos pontos que $\gamma(t)$.

A variação infinitesimal de $\gamma(t)$ pode então ser obtida a partir de

$$\delta\gamma(t) = \frac{d}{d\epsilon}\bigg|_{\epsilon=0} \gamma^\epsilon(t) \in T_{\gamma(t)}Q. \tag{B.14}$$

em que $T_{\gamma(t)}Q$ corresponde ao espaço tangente de Q na configuração $\gamma(t)$. Com isso, tem-se o ferramental matemático necessário para obter $\delta \boldsymbol{R}$ e $\delta \boldsymbol{p}$.

Com o intuito de encontrar a variação infinitesimal de $\delta \boldsymbol{R}$, pode-se definir uma variação de um elemento $R \in SO(3)$ a partir de um mapeamento exponencial:

$$\boldsymbol{R}^\epsilon = \boldsymbol{R} exp(\epsilon \hat{\boldsymbol{\eta}}), \tag{B.15}$$

em que $\hat{\boldsymbol{\eta}} \in \mathfrak{so}(3)$. Neste caso, $\boldsymbol{\eta}$ pode ser interpretado como um vetor que define o eixo de rotação e ϵ como uma variação de ângulo em torno deste eixo. A variação infinitesimal então é obtida a partir de (B.14) e o resultado é dado por

$$\delta \boldsymbol{R} = \boldsymbol{R}\hat{\boldsymbol{\eta}} \in T_{\boldsymbol{R}}SO(3). \tag{B.16}$$

Enquanto $\delta\boldsymbol{R}$ por si só define uma variação em $SO(3)$, pode-se, a partir de (B.16), obter uma relação com uma variação em um espaço semelhante ao Euclidiano sabendo que $\eta \in \mathbb{R}^3$.

Um procedimento semelhante pode ser feito para definir $\delta\boldsymbol{p}$, considerando que $\forall \boldsymbol{p}_1, \boldsymbol{p}_2 \in \mathbb{S}^2, \exists \boldsymbol{R}$ tal que $\boldsymbol{p}_2 = \boldsymbol{R}\boldsymbol{p}_1$. Portanto, pode-se utilizar uma variação de $\boldsymbol{R}$ para se obter uma variação de $\boldsymbol{p}$, da seguinte maneira,

$$\boldsymbol{p}^{\epsilon} = exp(\epsilon\hat{\boldsymbol{\xi}})\boldsymbol{p}, \tag{B.17}$$

em que $\xi \in \mathbb{R}^3$. A variação infinitesimal obtida a partir de (B.14) é

$$\delta \boldsymbol{p} = \hat{\boldsymbol{\xi}}\boldsymbol{p} \in T_{\boldsymbol{p}}S^2. \tag{B.18}$$

É importante mencionar que $\hat{\boldsymbol{\xi}}\boldsymbol{p}$ é equivalente a $\boldsymbol{\xi} \times \boldsymbol{p}$. A interpretação geométrica deste resultado é que uma variação infinitesimal de um vetor $\boldsymbol{p} \in \mathbb{S}^2$ resulta em um vetor normal a $\boldsymbol{p}$ causado por um vetor velocidade angular ξ.

Em resumo, pode-se definir as variações $\delta\boldsymbol{R}$ e $\delta\boldsymbol{p}$ como:

$$\delta \boldsymbol{p} = \boldsymbol{\xi} \times \boldsymbol{p} \in T_p\mathbb{S}^2, \qquad \text{tal que } \boldsymbol{\xi} \in \mathbb{R}^3, \boldsymbol{\xi} \cdot \boldsymbol{p} = 0, \tag{B.19}$$

$$\delta \boldsymbol{R} = \boldsymbol{R}\hat{\boldsymbol{\eta}} \in T_R SO(3), \quad \text{tal que } \boldsymbol{\eta} \in \mathbb{R}^3, \hat{\boldsymbol{\eta}} \in \mathfrak{so}(3) \tag{B.20}$$

e então obter suas derivações:

$$\delta \dot{\boldsymbol{p}} = \dot{\boldsymbol{\xi}} \times \boldsymbol{p} + \boldsymbol{\xi} \times \dot{\boldsymbol{p}}, \tag{B.21}$$

$$\begin{aligned}
\delta \dot{\boldsymbol{R}} &= \dot{\boldsymbol{R}}\hat{\boldsymbol{\eta}} + \boldsymbol{R}\dot{\hat{\boldsymbol{\eta}}}, \\
\delta \hat{\boldsymbol{\Omega}} &= \delta(\boldsymbol{R}^T\dot{\boldsymbol{R}}), \\
&= \delta\boldsymbol{R}^T\dot{\boldsymbol{R}} + \boldsymbol{R}^T\delta\dot{\boldsymbol{R}}, \\
&= (\boldsymbol{R}\hat{\boldsymbol{\eta}})^T\dot{\boldsymbol{R}} + \boldsymbol{R}^T(\dot{\boldsymbol{R}}\hat{\boldsymbol{\eta}} + \boldsymbol{R}\dot{\hat{\boldsymbol{\eta}}}), \\
&= \hat{\boldsymbol{\eta}}^T\boldsymbol{\Omega} + \hat{\boldsymbol{\Omega}}\hat{\boldsymbol{\eta}} + \dot{\hat{\boldsymbol{\eta}}}, \\
&= \widehat{\hat{\boldsymbol{\Omega}}\boldsymbol{\eta}} + \dot{\hat{\boldsymbol{\eta}}}, \\
\delta \boldsymbol{\Omega} &= \hat{\boldsymbol{\Omega}}\boldsymbol{\eta} + \dot{\boldsymbol{\eta}}.
\end{aligned} \tag{B.22}$$

Com as variações obtidas nas equações (B.19) a (B.22), as integrais em (B.13) podem ser reescritas em termos de $\delta \boldsymbol{x}_L$, $\boldsymbol{\xi}$ e $\boldsymbol{\eta}$ que são variações no $\mathbb{R}^3$. Em seguida, é necessário realizar uma etapa de integração por partes para eliminar os termos derivativos das variações e uma etapa de substituição para simplificação das expressões. Este é um processo bastante mecânico. Deste modo, os detalhes estão descritos no Apêndice B.

O resultado da simplificação de (B.13) é dado por

$$\begin{aligned}\delta S = & \int_{t_1}^{t_2} \Big(\big(m_Q l \ddot{\boldsymbol{p}} - (m_Q + m_L)\ddot{\boldsymbol{x}}_L + f\boldsymbol{R}\boldsymbol{e}_3 - (m_Q + m_L) g \boldsymbol{e}_3 \big) \cdot \delta \boldsymbol{x}_L \Big) dt \\ & + \int_{t_1}^{t_2} \Big(\big(\boldsymbol{p} \times (m_Q g l \boldsymbol{e}_3 - f l \boldsymbol{R}\boldsymbol{e}_3 + m_Q l \ddot{\boldsymbol{x}}_L - m_Q l^2 \ddot{\boldsymbol{p}}) \big) \cdot \boldsymbol{\xi} \Big) dt \\ & + \int_{t_1}^{t_2} \Big((-\boldsymbol{\Omega} \times \boldsymbol{J}_Q \boldsymbol{\Omega} - \boldsymbol{J}_Q \dot{\boldsymbol{\Omega}} + \boldsymbol{M}) \cdot \eta \Big) dt = 0 \end{aligned} \tag{B.23}$$

Em seguida, ao igualar os termos das integrais a zero para se obter as equações de movimento do sistema quadricóptero-carga, obtém-se

$$m_Q l \ddot{\boldsymbol{p}} - (m_Q + m_L)\ddot{\boldsymbol{x}}_L + f\boldsymbol{R}\boldsymbol{e}_3 - (m_Q + m_L) g \boldsymbol{e}_3 = 0, \tag{B.24}$$

$$\boldsymbol{p} \times (m_Q g \boldsymbol{e}_3 - f \boldsymbol{R}\boldsymbol{e}_3 + m_Q \ddot{\boldsymbol{x}}_L - m_Q l \ddot{\boldsymbol{p}}) = 0, \tag{B.25}$$

$$-\boldsymbol{\Omega} \times \boldsymbol{J}_Q \boldsymbol{\Omega} - \boldsymbol{J}_Q \dot{\boldsymbol{\Omega}} + \boldsymbol{M} = 0. \tag{B.26}$$

As equações diferenciais (B.24) e (B.25) são decompostas em equações diferenciais de ordem menor para se obter uma representação no espaço de estados, resultando em

$$\dot{\boldsymbol{x}}_L = \boldsymbol{v}_L, \tag{B.27}$$

$$(m_L + m_Q)(\dot{\boldsymbol{v}}_L + g\boldsymbol{e}_3) = (\boldsymbol{p} \cdot f\boldsymbol{R}\boldsymbol{e}_3 - m_Q l(\dot{\boldsymbol{p}} \cdot \dot{\boldsymbol{p}}))\boldsymbol{p}, \tag{B.28}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{\omega} \times \boldsymbol{p}, \tag{B.29}$$

$$m_Q l \dot{\boldsymbol{\omega}} = -\boldsymbol{p} \times f\boldsymbol{R}\boldsymbol{e}_3, \tag{B.30}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\Omega}}, \tag{B.31}$$

$$\boldsymbol{J}_Q \dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \boldsymbol{J}_Q \boldsymbol{\Omega} = \boldsymbol{M}, \tag{B.32}$$

em que as equações de (B.27) a (B.30) representam as dinâmicas da posição e orientação da carga e as equações (B.31) e (B.32) representam a dinâmica da orientação do quadricóptero. Vale enfatizar que o estado do sistema é dado por

$$\boldsymbol{\mathcal{X}} = [\boldsymbol{x}_L, \boldsymbol{v}_L, \boldsymbol{p}, \boldsymbol{\omega}, \boldsymbol{R}, \boldsymbol{\Omega}]. \tag{B.33}$$

APÊNDICE C – Funções de erro de rastreamento

A equações de realimentação dos estados nos subsistemas de controle precisam conter algum tipo de métrica de erro. Em um controle geométrico, a definição de erro de configuração ou rastreamento, caso a configuração desejada varie no tempo, é feita no *manifold* em que se encontra o vetor de configuração. No espaço euclidiano, a função de erro de rastreamento pode ser definida como

$$\Psi_{\mathbb{R}^3}(\boldsymbol{x}, \boldsymbol{x}_d) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_d) \cdot (\boldsymbol{x} - \boldsymbol{x}_d), \tag{C.1}$$

em que $\boldsymbol{x}$ é a configuração de um sistema e $\boldsymbol{x}_d$ é a configuração desejada, ambos pertencendo ao $\mathbb{R}^3$. Assim, os erros utilizados em sistemas de controle atuando no $\mathbb{R}^3$ são obtidos a partir do diferencial de $\Psi_{\mathbb{R}^3}(\boldsymbol{x})$ da seguinte forma

$$\boldsymbol{e}_x = d_1\Psi_{\mathbb{R}^3}(\boldsymbol{x}, \boldsymbol{x}_d) = (\boldsymbol{x} - \boldsymbol{x}_d), \tag{C.2}$$

em que $d_1\Psi_{\mathbb{R}^3}$ representa o diferencial de $\Psi_{\mathbb{R}^3}$ com relação ao primeiro parâmetro $\boldsymbol{x}$, e a derivada do tempo é

$$\dot{\boldsymbol{e}}_x = (\dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_d). \tag{C.3}$$

Sabendo que $\boldsymbol{x}$ representa a posição, a equação (C.3) pode ser reescrita em termos de velocidade

$$\boldsymbol{e}_v = (\boldsymbol{v} - \boldsymbol{v}_d). \tag{C.4}$$

Com (C.2) e (C.4), é possível definir uma equação de controle por realimentação no $\mathbb{R}^3$.

Resta definir as funções de erro de rastreamento para o S^2 e o $SO(3)$. O processo de obtenção do diferencial de cada função envolve um conjunto de conhecimentos que saem do escopo da dissertação, mas são apresentadas com detalhes na seção 11.1.3 para o S^2 e na seção 11.4 para o $SO(3)$ em Bullo e Lewis (2004). Nesta seção, são apenas apresentadas as equações que dão suporte às definições para o projeto dos controladores geométricos.

A função erro de rastreamento para a orientação da carga, que evolui no S^2, é então definida como

$$\Psi_{S^2}(\boldsymbol{p}, \boldsymbol{p}_d) = 1 - \boldsymbol{p} \cdot \boldsymbol{p}_d, \tag{C.5}$$

em que $\boldsymbol{p}$ é a configuração de orientação da carga e $\boldsymbol{p}_d$ é a configuração desejada.

Como o produto escalar entre dois vetores unitários é dado por $cos(\phi)$, em que ϕ é o ângulo entre os vetores, a função de erro de rastreamento pode então ser reescrita como $\Psi_{S^2}(\boldsymbol{p}, \boldsymbol{p}_d) = 1 - cos(\phi)$. Esta função possuir valores entre 0, quando as duas configurações forem iguais, e 2, quando as configurações possuem sentidos opostos.

Com isso, é feito um procedimento semelhante a (C.2) e (C.3) em que os erros que são utilizados nas equações de controle situam-se no espaço tangente. Por isso, encontra-se o diferencial de $\Psi_{S^2}(\boldsymbol{p}, \boldsymbol{p}_d)$ que é dado por

$$d_1\Psi_{S^2}(\boldsymbol{p}, \boldsymbol{p}_d) = \hat{\boldsymbol{p}}^2\boldsymbol{p}_d, \tag{C.6}$$

em que $d_1\Psi_{S^2}(\boldsymbol{p}, \boldsymbol{p}_d)$ representa o diferencial de Ψ_{S^2} com relação ao primeiro parâmetro $\boldsymbol{p}$.

Então, a função de erro de orientação da carga $\boldsymbol{e_p}$ é definida como

$$\boldsymbol{e_p} = \hat{\boldsymbol{p}}^2\boldsymbol{p}_d. \tag{C.7}$$

O processo de obtenção do erro de velocidade $\boldsymbol{e_{\dot{p}}}$ é mais complexo que o realizado no $\mathbb{R}^3$, em que os vetores são comparados diretamente. Isto ocorre porque cada vetor pertence a espaços tangentes diferentes. Então Bullo e Lewis (2004, p. 545) apresenta o erro de velocidade a partir de um mapeamento que transporta um vetor de um espaço para o outro como

$$\boldsymbol{e_{\dot{p}}} = \dot{\boldsymbol{p}} - (\boldsymbol{p}_d \times \dot{\boldsymbol{p}}_d) \times \boldsymbol{p}. \tag{C.8}$$

Este processo é semelhante para a definição de erro de rastreamento da a dinâmica da orientação do quadricóptero, que evolui no $SO(3)$. A função erro de rastreamento é então definida como

$$\Psi_{\boldsymbol{R}}(\boldsymbol{R}, \boldsymbol{R}_d) = \frac{1}{2}\Big[I - \boldsymbol{R}_d^T\boldsymbol{R}\Big], \tag{C.9}$$

em que $\boldsymbol{R}$ é a configuração de orientação do quadricóptero e $\boldsymbol{R}_d$ é a configuração desejada.

Quando $\boldsymbol{R} = \boldsymbol{R}_d$, $\Psi_{\boldsymbol{R}} = 0$ e o máximo desta função é 2, que ocorre quando $\boldsymbol{R}$ está a 180° de diferença com relação à orientação desejada. Estas características são bastante semelhantes a (C.5). Deste modo, a região dada por $\Psi_{\boldsymbol{R}} < 2$ cobre quase todo $SO(3)$. Esta informação é usada para definir a região de atração do controlador de orientação na seção 3.3.1.

A variação de $\Psi_{\boldsymbol{R}}$ que define o erro de orientação utilizado no controlador é obtida em Bullo e Lewis (2004, p. 553) como

$$\boldsymbol{e_R} = skew(\boldsymbol{R}_d^T\boldsymbol{R})^\vee, \tag{C.10}$$

em que $skew(\boldsymbol{A})$ representa a parte anti-simétrica de $\boldsymbol{A}$ para qualquer matriz quadrada real dada por

$$skew(\boldsymbol{A}) = \frac{1}{2}(\boldsymbol{A} - \boldsymbol{A}^T). \tag{C.11}$$

Fazendo $A = \boldsymbol{R}_d^T \boldsymbol{R}$, segue que

$$\boldsymbol{e_R} = (\boldsymbol{R}_d^T \boldsymbol{R} - \boldsymbol{R}^T \boldsymbol{R}_d)^\vee. \tag{C.12}$$

E o erro de velocidade angular é obtido partindo do devido mapa de transporte que projeta o vetor velocidade no espaço tangente a $\boldsymbol{R}_d$ pra o espaço tangente a $\boldsymbol{R}$. Este processo é descrito em Bullo e Lewis (2004, p. 554) e o resultado é

$$\boldsymbol{e_\Omega} = \boldsymbol{\Omega} - \boldsymbol{R}^T \boldsymbol{R}_d \boldsymbol{\Omega}_d., \tag{C.13}$$

em que $\boldsymbol{\Omega}$ é a velocidade angular do quadricóptero com relação ao sistema de coordenadas anexado ao corpo e $\boldsymbol{\Omega}_d$ é a velocidade angular desejada, no mesmo sistema de coordenadas.

Com as devidas métricas de erro de configuração para cada *manifold* do sistema, é possível então projetar as equações dos controladores de orientação do quadricóptero, em seguida, de orientação da carga e, por fim, de posição da carga.