

---

Hybrid qualitative state plan problem e o  
planejamento de missão com VANTs

*Márcio da Silva Arantes*

---



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

**Márcio da Silva Arantes**

## Hybrid qualitative state plan problem e o planejamento de missão com VANTs

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Claudio Fabiano Motta Toledo

**USP – São Carlos**  
**Outubro de 2017**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados fornecidos pelo(a) autor(a)

A634h Arantes, Márcio da Silva  
Hybrid qualitative state plan problem e  
o planejamento de missão com VANTs / Márcio  
da Silva Arantes; orientador Claudio Fabiano  
Motta Toledo. - São Carlos - SP, 2017.  
157 p.

Tese (Doutorado - Programa de Pós-Graduação em  
Ciências de Computação e Matemática Computacional)  
- Instituto de Ciências Matemáticas e de Computação,  
Universidade de São Paulo, 2017.

1. Planejamento de Missão. 2. Veículos Aéreos  
Não Tripulados. 3. Programação Linear Inteira-Mista.  
4. Computação Evolutiva . I. Toledo, Claudio Fabiano  
Motta, orient. II. Título.



**Márcio da Silva Arantes**

Hybrid qualitative state plan problem and mission planning  
with UAVs

Doctoral dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP, in partial fulfillment of the requirements for the degree of the Doctorate Program in Computer Science and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Claudio Fabiano Motta Toledo

**USP – São Carlos**  
**October 2017**



*Dedico este trabalho aos meus pais e aos meus avós,  
pois com muita luta e sacrifícios batalharam e, assim, conseguiram  
dar-me melhores oportunidades, as quais não tiveram.  
Sempre amarei vocês!*



# AGRADECIMENTOS

---

---

Ao meu orientador, prof. Claudio Fabiano Motta Toledo, meus sinceros agradecimentos pelo empenho e dedicação demonstrados. Com sua orientação descobri minha vocação para a pesquisa científica

A todos os professores, pelo entusiasmo com o qual ministraram as disciplinas cursadas, pois contribuíram diretamente para minha formação.

Ao meu irmão e melhor amigo, Jesimar da Silva Arantes, por estar sempre por perto e compartilhar tudo que aprende comigo.

Aos colegas do Laboratório de Computação Reconfigurável, pela ótima convivência e amizade durante o desenvolvimento deste trabalho.

À FAPESP pelo apoio financeiro concedido, sem o qual, não seria possível a realização deste trabalho (processo 2014/11331-0).

À Deus pela caminhada e por toda a alegria nas experiências que vivi.



*“Não se espante com a altura do vôo. Quanto mais alto, mais longe do perigo.  
Quanto mais você se eleva, mais tempo há de reconhecer uma pane.  
É quando se está próximo do solo que se deve desconfiar”*  
*(Santos Dumont)*





# RESUMO

ARANTES, M. S. **Hybrid qualitative state plan problem e o planejamento de missão com VANTs**. 2017. 157 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

O presente documento tem por objetivo apresentar a tese desenvolvida no Programa de Doutorado em Ciência da Computação e Matemática Computacional do ICMC/USP. O tema da tese busca avançar o estado da arte ao resolver os problemas de escalabilidade e representação presentes em algoritmos de planejamento para missões com Veículos Aéreos Não Tripulados (VANTs). Técnicas baseadas em programação matemática e computação evolutiva são propostas. Artigos foram publicados, submetidos ou se encontram em fase final de elaboração. Esses trabalhos reportam os avanços mais significativos obtidos na representação e escalabilidade deste problema. Os planejadores de missão trabalhados na tese lidam com problemas estocásticos em ambientes não convexos, onde os riscos de colisão ou falhas no planejamento da missão são tratados e limitados a um valor tolerado. Os avanços na representação permitiram solucionar violações nos riscos presentes na modelagem original, além de tornar os modelos mais realistas ao incorporar aspectos como efeitos da resistência do ar. Para isso, técnicas eficientes de modelagem matemática permitiram avançar de um modelo de Programação Não-Linear Inteira Mista (PNLIM), originalmente proposto na literatura, para um problema de Programação Linear Inteira Mista (PLIM). A modelagem como um PLIM levou à resolução do problema de forma mais eficiente através do algoritmo *branch-and-cut*. As novas representações propostas resultaram em melhorias na escalabilidade, solucionando problemas mais complexos em um tempo computacional menor. Além disso, os avanços em escalabilidade mostraram-se mais efetivos quando técnicas combinando programação matemática e metaheurísticas foram aplicadas ao problema.

**Palavras-chave:** Planejamento de Missão, Veículos Aéreos Não Tripulados, Programação Linear Inteira-Mista, Computação Evolutiva .



# ABSTRACT

ARANTES, M. S. **Hybrid qualitative state plan problem and mission planning with UAVs**. 2017. 157 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

This paper aims to present the thesis developed in the Doctoral Program in Computer Science and Computational Mathematics of the ICMC / USP. The thesis theme seeks to advance the state of the art by solving the problems of scalability and representation present in mission planning algorithms for Unmanned Aerial Vehicle (UAV). Techniques based on mathematical programming and evolutionary computation are proposed. Articles have been published, submitted or they are in final stages of preparation. These studies report the most significant advances in the representation and scalability of this problem. Mission planners worked on the thesis deal with stochastic problems in non-convex environments, where collision risks or failures in mission planning are treated and limited to a tolerated value. The advances in the representation allowed to solve violations in the risks present in the original literature modeling, besides making the models more realistic when incorporating aspects such as effects of the air resistance. Efficient mathematical modeling techniques allowed to advance from a Mixed Integer Nonlinear Programming (MINLP) model, originally proposed in the literature, to a Mixed Integer Linear Programming (MILP) problem. Modeling as a MILP led to problem solving more efficiently through the branch-and-cut algorithm. The proposed new representations resulted in improvements from scalability, solving more complex problems within a shorter computational time. In addition, advances in scalability are even more effective when techniques combining mathematical programming and metaheuristics have been applied to the problem.

**Keywords:** Mission Planning, Unmanned Aerial Vehicle, Integer-Mixed Linear Programming, Evolutionary Computation .



# SUMÁRIO

---

---

1	<b>INTRODUÇÃO</b>	19
1.1	Motivação	19
1.2	Descrição do problema	19
1.3	Desafios e objetivos	21
1.4	Plataforma Ararinha	22
1.5	Organização do texto	24
2	<b>TRABALHOS RELACIONADOS</b>	25
3	<b>FUNDAMENTAÇÃO MATEMÁTICA</b>	33
3.1	Conectivos lógicos	33
3.2	Modelando implicações tautológicas	38
4	<b>CHANCE-CONSTRAINED PATH-PLANNING PROBLEM</b>	43
4.1	Introdução	43
4.2	Descrição do problema	44
4.3	Modelo determinístico para o problema	46
4.4	Discussão a respeito das hipóteses do modelo	50
4.5	Modelo linear para o problema	52
5	<b>CHANCE-CONSTRAINED QUALITATIVE STATE PLAN PROBLEM</b>	55
5.1	Introdução	55
5.2	Descrição do problema	56
5.3	Codificando o problema	60
5.3.1	<i>Codificação previamente estabelecida na literatura</i>	60
5.3.2	<i>Codificando o scheduling</i>	61
5.4	Decodificando o problema em restrições lineares	62
5.4.1	<i>Modelo completo do CCQSP Planning Problem</i>	68
5.5	Melhorias do modelo	71
5.5.1	<i>Equações de dinâmica com efeito de arrasto</i>	71
5.5.2	<i>Restrições de risco reduzidas</i>	74
5.5.3	<i>Adicionando restrições para fortalecer o modelo</i>	76
5.5.3.1	<i>O episódio estará ativo em algum instante ( Caso 1 )</i>	76
5.5.3.2	<i>Existem episódios Remain-In que estarão ativos sempre ( Caso 2 )</i>	78

5.5.3.3	<i>Regiões de um ponto de passagem excluem-se mutuamente ( Caso 3 )</i> . . . .	78
5.5.3.4	<i>Hiperplanos de um obstáculo excluem-se mutuamente ( Caso 4 )</i> . . . . .	79
5.6	<b>Análise de complexidade</b> . . . . .	79
6	<b>HYBRID CHANCE-CONSTRAINED QUALITATIVE STATE PLAN</b>	
	<b>PROBLEM</b> . . . . .	93
6.1	<b>Introdução</b> . . . . .	93
6.2	<b>Descrição do problema</b> . . . . .	94
6.3	<b>Modelo para o HCCQSP</b> . . . . .	99
6.3.1	<i>Restrições modelando o comportamento das ações</i> . . . . .	100
6.3.2	<i>Restrições modelando o planejamento da missão</i> . . . . .	102
7	<b>RESULTADOS</b> . . . . .	105
7.1	<b>Resultados para o CCPP</b> . . . . .	106
7.1.1	<i>Metodologia e cenários</i> . . . . .	106
7.1.2	<i>Resultados de escalabilidade para mapas 2D</i> . . . . .	108
7.1.3	<i>Resultados de escalabilidade para mapas 3D</i> . . . . .	110
7.1.4	<i>Resultados em simulação com FlightGear</i> . . . . .	112
7.1.5	<i>Avaliando o desvio de obstáculos</i> . . . . .	114
7.1.6	<i>Abordagens com computação evolutiva</i> . . . . .	117
7.1.7	<i>Resultados para processamento embarcado</i> . . . . .	119
7.2	<b>Resultados para o CCQSP</b> . . . . .	122
7.2.1	<i>Comparação do pSulu e propostas de PLIM</i> . . . . .	123
7.2.2	<i>Resultados das melhorias propostas</i> . . . . .	124
7.2.3	<i>Resultados de escalabilidade</i> . . . . .	128
7.3	<b>Resultados para o HCCQSP</b> . . . . .	131
8	<b>CONCLUSÃO</b> . . . . .	135
	<b>REFERÊNCIAS</b> . . . . .	137
ANEXO A	<b>MELHORIAS PARA O CCPP</b> . . . . .	143
ANEXO B	<b>MELHORIAS PARA O CCQSP</b> . . . . .	145
ANEXO C	<b>ALGORITMOS EVOLUTIVOS PARA O CCPP</b> . . . . .	147
ANEXO D	<b>ABORDAGENS PARA O DESVIO DE OBSTÁCULOS</b> . . . . .	149
ANEXO E	<b>GARANTIAS NO RISCO DE COLISÃO</b> . . . . .	151
ANEXO F	<b>POUSO EM CASO DE FALHAS - CONGRESSO</b> . . . . .	153

<b>ANEXO G</b>	<b>POUSO EM CASO DE FALHAS - REVISTA . . . . .</b>	<b>155</b>
<b>ANEXO H</b>	<b>ARQUITETURA EMBARCADA . . . . .</b>	<b>157</b>





---

# INTRODUÇÃO

---

## 1.1 Motivação

Considerando as perspectivas tecnológicas atuais, é esperado que sistemas autônomos estejam cada vez mais presentes no cotidiano, auxiliando nas mais diversas atividades humanas. A ciência, de forma geral, tem sido a base para isso, ao definir os formalismos matemáticos necessários para avançar pesquisas em busca de construir sistemas robóticos teleoperados e semi-autônomos. O ritmo atual da evolução tecnológica viabiliza, cada vez mais, a construção de computadores embarcados com baixo custo dos componentes. Institutos de pesquisa, universidades e empresas avançam em direção a uma maior autonomia destes sistemas, lançando componentes novos ou melhorados para sensoriamento, comunicação e processamento embarcado. A cada nova versão destes componentes, propostas de melhor performance surgem e os custos financeiros de fabricação e o consumo energético dos sistemas embarcados diminuem. Agências militares e de segurança pública investem em sistemas que utilizam Veículos Aéreos Não Tripulados (VANTs) em operações de vigilância. VANTs costumam ser teleoperados ou seguem uma rota estabelecida e gerenciada por um piloto automático embarcado na aeronave. O presente projeto de doutorado está focado em contribuir para o planejamento automático de rota para VANTs. Esse planejamento será chamado aqui de planejamento de missão, pois tal planejamento vai muito além de uma simples rota, como estabelecido em trabalhos anteriores ([BLACKMORE; ONO; WILLIAMS, 2011](#); [ONO; WILLIAMS; BLACKMORE, 2013](#); [LI, 2010](#)).

## 1.2 Descrição do problema

Vamos supor inicialmente um cenário de missão como o ilustrado na Figura 1, onde um VANT inicia em um aeroporto, voa até uma região previamente definida para tirar fotografias e, terminando a missão, retorna ao aeroporto. Durante o voo, o VANT poderá

sobrevoar ou contornar regiões montanhosas para completar a missão. Considerando um voo autônomo, torna-se relevante que o VANT não sobrevoe aeroportos civis, áreas militares e fronteiras políticas, áreas que são denominadas restritas ao voo, e regiões povoadas. Cabe ressaltar que pode haver regiões restritas ao voo devido a condições climáticas adversas como tempestades, além da constante preocupação do VANT em evitar colisões com outras aeronaves que estejam voando no mesmo cenário.

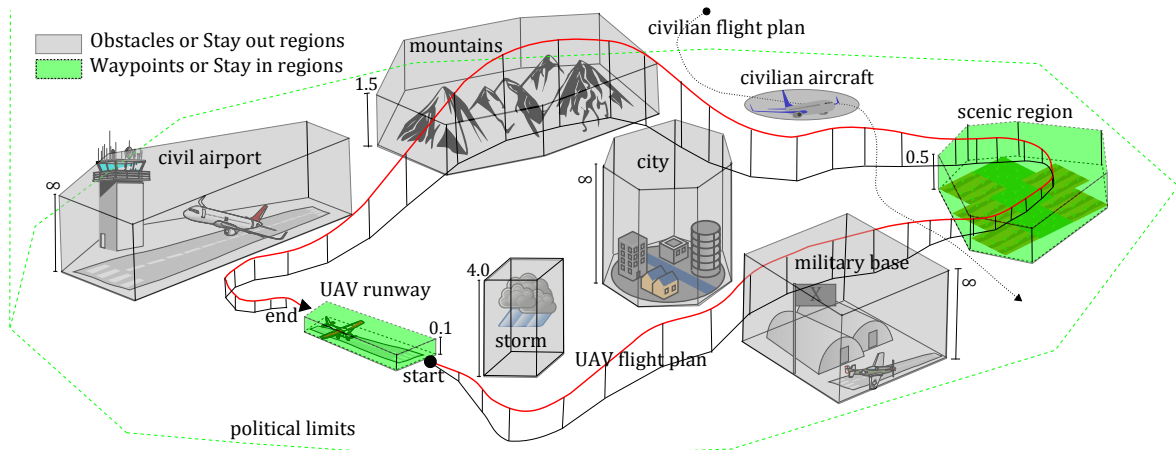


Figura 1 – Descrição do problema

O planejador para essa missão deve levar em consideração a incerteza a respeito do ambiente que pode ser causada por distúrbios durante o voo, ventos, localização imprecisa e erros de modelagem da própria dinâmica de voo. Assim, o planejador de missão deve resolver um problema estocástico e planejar uma rota de voo segura dentro de uma margem de risco, evitando colidir com os obstáculos (*Stay out regions*) pelo caminho e respeitando os pontos de passagem (*Stay in regions*). A missão planejada deve otimizar algum critério como consumo de combustível, distância percorrida ou tempo da missão, sem violar a margem de risco especificada para a missão.

No exemplo apresentado na Figura 1, alguns obstáculos como cidades, aeroportos civis e bases militares são consideradas como possuindo altura ilimitada, não permitindo assim que o VANT sobrevoe esses obstáculos em hipótese alguma. Montanhas, florestas e tempestades são também consideradas obstáculos, mas com alturas limitadas, assim se torna possível voar sobre esses obstáculos. Também são definidas regiões de interesse (*Stay in regions*), onde o VANT deverá executar alguma tarefa específica como tirar fotos, pousar e decolar.

O problema lida com a incerteza do sistema, em que a dinâmica do veículo é representada como um sistema dinâmico estocástico seguindo a equação 1.1, onde  $\mathbf{x}_t \in \mathbb{X} \subseteq \mathbb{R}^n$  e  $\mathbf{u}_t \in \mathbb{U} \subseteq \mathbb{R}^m$  são variáveis aleatórias representando os estados do sistema e controles aplicados a cada instante de tempo  $t$ . As matrizes  $A$  e  $B$  definem a transição

dos estados  $\mathbf{x}_t$  para  $\mathbf{x}_{t+1}$  dados os controles  $\mathbf{u}_t$  aplicados.

$$\mathbf{x}_{t+1} = A\mathbf{x}_t + B\mathbf{u}_t + \omega_t \quad \forall(t) \quad (1.1)$$

As incertezas estão relacionadas com a imprecisão da posição inicial ( $\mathbf{x}_0$ ), fatores externos que agem sobre a dinâmica do veículo e erros proveniente de a uma modelagem aproximada da dinâmica do veículo. A posição inicial e o estado inicial são assumidos como seguindo uma distribuição Gaussiana com valor esperado  $\bar{\mathbf{x}}_0$  e matriz de covariância  $\Sigma_{x_0}$ , assim  $\mathbf{x}_0 \sim \mathcal{N}(\bar{\mathbf{x}}_0, \Sigma_{x_0})$ . As incertezas provocadas por fatores externos e erros de modelagem são adicionadas ao sistema a cada instante de tempo  $t$  com um ruído aditivo  $\omega_t$ , considerando também uma distribuição Gaussiana com média zero e matriz de covariância  $\Sigma_{\omega_t}$  onde  $\omega_t \sim \mathcal{N}(0, \Sigma_{\omega_t})$ . Um ruído aditivo faz sentido para aplicações em VANTs, pois distúrbios independentes de estado como vento e sensoriamento impreciso estão sempre presentes.

### 1.3 Desafios e objetivos

Por se tratar de um problema estocástico e não convexo, viabilizar o planejamento de missão para VANTs exige avançar em dois aspectos ainda não resolvidos: representação e escalabilidade. A representação consiste em encontrar soluções viáveis para serem aplicadas ao mundo real. A escalabilidade é a capacidade de resolver cenários com missões mais complexas dentro de um tempo de computação viável. Avançar o estado da arte em escalabilidade e representação não é uma tarefa fácil, pois cenários mais representativos e complexos, geralmente, implicam em maior esforço computacional. Isso se justifica pelo estado da arte dos algoritmos de planejamento de missão considerados neste trabalho. Para alcançar esses objetivos, esta tese propõe um conjunto de formalismos matemáticos capazes de tornar os cenários de planejamento de missão mais representativos, além de desenvolver a matemática dos algoritmos de planejamento para lidar de forma mais eficiente com a execução das missões. Assim, a tese busca avançar mais especificamente no estado da arte dos planejadores de missão apresentados em (BLACKMORE; ONO; WILLIAMS, 2011; ONO; WILLIAMS; BLACKMORE, 2013; LI, 2010).

A Figura 2 apresenta em linhas gerais as contribuições alcançadas nesta tese, onde duas contribuições principais se destacam. A primeira contribuição está em melhorar os resultados de representação e escalabilidade presentes em três problemas distintos: CCP, CCQSP e HQSP. A segunda contribuição é a combinação das características dos três problemas em um problema completo, chamado aqui de HCCQSP, que considera para do planejamento da missão a resolução de um problema estocástico com *scheduling* e ações híbridas. A estocasticidade se dá pela solução de um problema em que é considerada as incertezas inerentes a um ambiente real. O *scheduling* está em tomar a decisão ótima em

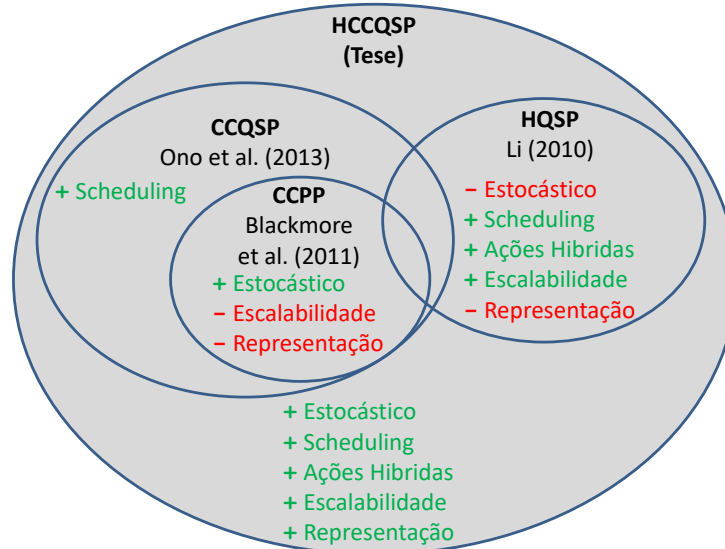


Figura 2 – Visão geral da tese e contribuições

relação aos agendamentos de tempo estabelecidos para o VANT executar tarefas definidas (*Stay in regions*). Por fim, ações híbridas permitem VANT tomar as decisões ótimas e de forma autônoma em relação as ações a serem executadas para o cumprimento da missão. Nesse caso, considera-se tanto ações discretas, como tirar foto, quanto ações contínuas, como voar. Nos Capítulos 4, 5 e 6 serão apresentados maiores detalhes sobre os problemas CCPP, CCQSP e HCCQSP respectivamente.

## 1.4 Plataforma Ararinha

Um dos objetivos da tese foi desenvolver os algoritmos planejadores de missão e resolver os problemas de escalabilidade e representação a ponto de serem embarcados dentro de arquiteturas de hardware utilizadas em VANTs como a aeronave Ararinha, desenvolvida no ICMC e descrita a seguir. A Figura 3 apresenta uma aeronave elétrica do tipo Ararinha (SCUSSEL, 2016), construída durante este projeto de doutorado.

Essa aeronave é capaz de realizar um voo autônomo, em que a rota é fornecida ao piloto automático. Abaixo segue uma descrição detalhada do que faz cada equipamento apresentado na Figura 3.

- (a) **telemetria na aeronave:** permite a aeronave receber, remotamente, configurações para o piloto automático e estabelecer, previamente, uma rota de voo definida por um conjunto de waypoints.
- (b) **GPS com IMU:** módulo contendo um Global Positioning System (GPS) e Inertial Measurement Unit (IMU) de alta precisão que fornecem ao piloto automático a posição e orientação da aeronave necessárias para navegação autônoma.

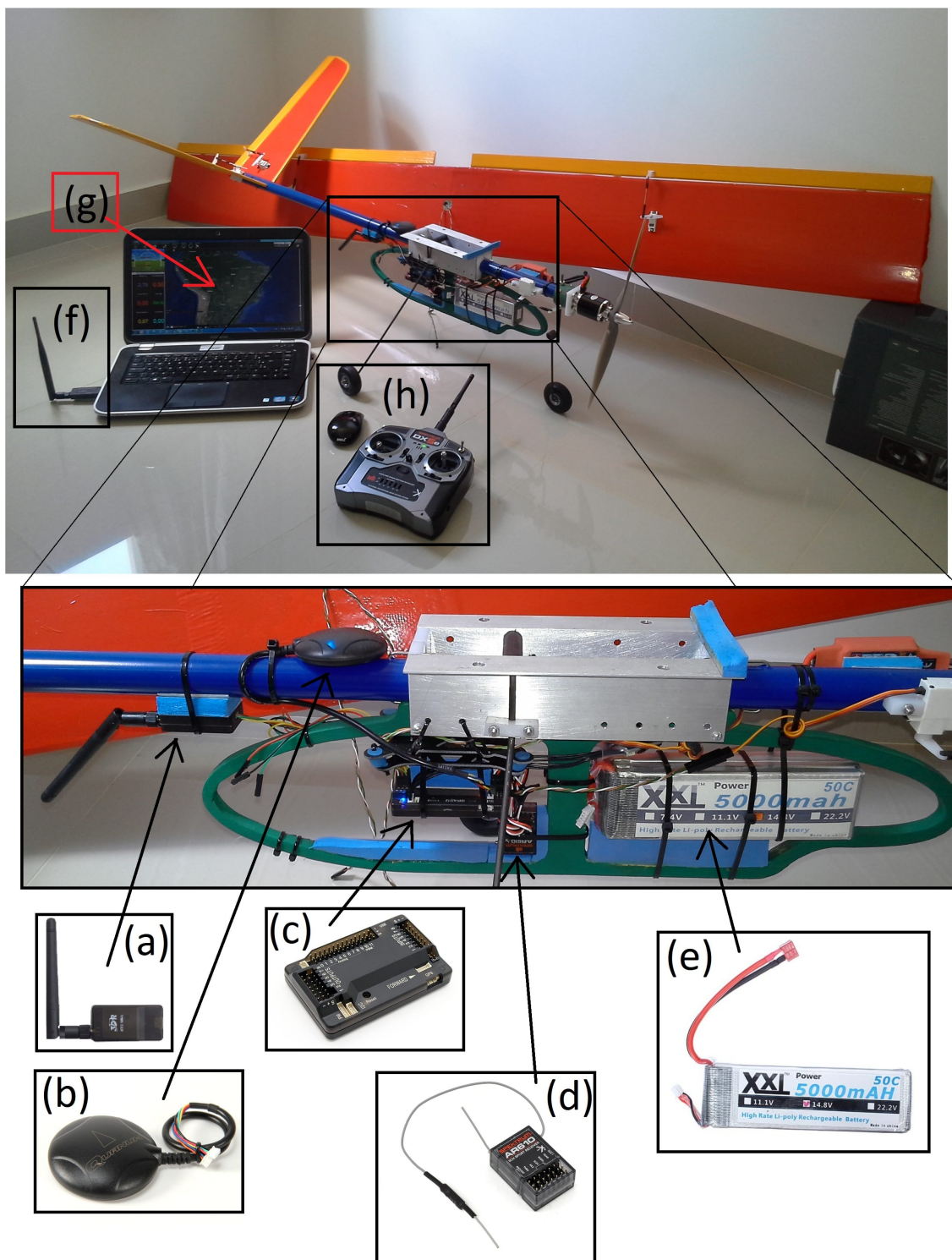


Figura 3 – Aeronave do tipo Ararinha

- (c) **piloto automático:** utilizado para navegação autônoma, capaz de seguir um conjunto de waypoints estabelecidos e executar algumas manobras básicas, como por exemplo: manter a altitude, voar em círculos e estabilizar a aeronave.
- (d) **receptor do rádio controle:** recebe comandos diretamente do controle manual do avião, utilizado em situações que exigem a intervenção humana. Atualmente, os

pousos e a decolagem da plataforma precisam ser executados no controle manual. Esse receptor precisa ser mantido na aeronave por segurança e é possível chavear facilmente entre os modos automático e manual.

- (e) **bateria recarregável:** a bateria alimenta o motor e todo hardware da aeronave.
- (f) **telemetria no computador:** permite configurar o piloto automático pelo computador utilizando o software Mission Planner ([ARDUPILOT, 2016](#)). Esse software é utilizado para enviar a rota para o piloto automático.
- (g) **computador:** um computador é utilizado para estabelecer uma rota por um algoritmo planejador de rota.
- (h) **rádio controle:** permite controlar o VANT diretamente ou passar a navegação para o piloto automático.

## 1.5 Organização do texto

- **Capítulo 2:** apresenta a revisão dos trabalhos relacionados, indicando as limitações das propostas existentes e explicitando de que maneira pretende-se avançar nesta tese.
- **Capítulo 3:** introduz a fundamentação matemática necessária ao entendimento desta tese, explicando como utilizar variáveis lógicas para a construção de modelos para PLIM de forma eficiente.
- **Capítulos 4, 5 e 6:** o problema estudado nesta tese é estabelecido a partir de outros problemas envolvendo o planejamento de missões. Os estudos conduzidos nesta tese procuram avançar aspectos relacionados a escalabilidade e representatividade também presente em tais problemas.
- **Capítulo 7:** relata as contribuições alcançadas na tese.
- **Capítulo 8:** apresenta as considerações finais sobre o trabalho, onde é sumarizado o que foi discutido neste texto.
- **Anexos A, B, C, D, E, F, G e H:** apresentam os artigos publicados, aceitos ou em fase final de elaboração que detalham os avanços obtidos durante o desenvolvimento da tese.



---

## TRABALHOS RELACIONADOS

---

As pesquisas com veículos autônomos tornaram-se cada vez mais relevantes para aplicações reais nos últimos anos (SALEEM; REHMANI; ZEDADALLY, 2015; YANG *et al.*, 2015; FREEMAN; FREELAND, 2015; BAYAT *et al.*, 2017). Há vários trabalhos sobre planejamento de trajetória para veículos autônomos que modelam estes como sistemas dinâmicos (SCHOUWENAARS *et al.*, 2001; RICHARDS; HOW, 2002; BLACKMORE; ONO, 2009; LI, 2010; BLACKMORE; ONO; WILLIAMS, 2011; ONO; WILLIAMS; BLACKMORE, 2013). Um sistema dinâmico descreve a transição do sistema para o estado atual a partir dos estados anteriores. Tratar o planejamento de trajetória considerando um sistema dinâmico e linear possibilita que o problema seja resolvido com o uso de modelos baseados em programação linear inteira mista (PLIM) (SCHOUWENAARS *et al.*, 2001; BLACKMORE; ONO; WILLIAMS, 2011; ONO; WILLIAMS; BLACKMORE, 2013; LI; WILLIAMS, 2011) ou programação quadrática inteira mista (PQIM) (LI, 2010) e (ONO; WILLIAMS; BLACKMORE, 2013). O horizonte de planejamento passa a ser dividido em um conjunto de passos e métodos de resolução baseados em técnicas como *branch-and-bound* podem ser empregados.

Os autores em (SCHOUWENAARS *et al.*, 2001) apresentam um modelo de programação linear inteira mista (PLIM) para o planejamento de trajetórias, visando o consumo ótimo de combustível para veículos autônomos. Esse modelo permitiu solucionar o problema na otimalidade utilizando solvers comerciais. A tese em (SCHOUWENAARS, 2005) introduz conceitos e algoritmos para o planejamento de trajetória considerando *Online Receding Horizon*, onde formulações do tipo PLIM são apresentadas com restrições para o desvio de obstáculos. O trabalho em (CULLIGAN, 2006) aplicou os conceitos em (SCHOUWENAARS *et al.*, 2001) em um ambiente 3D. Em (KUWATA, 2003) é modelado o desvio de obstáculos com uma margem de segurança onde a ideia principal era expandir o tamanho dos obstáculos e assim construir a trajetória sem colisão dentro de uma distância segura.

(BLACKMORE; ONO, 2009) definem uma melhoria no tratamento da segurança, com um modelo convexo para o planejamento de trajetória capaz de garantir que um veículo autônomo mantivesse uma distância de segurança em relação aos limites da região convexa onde se encontrava. Os autores trataram a incerteza existente através de um modelo estocástico, onde uma probabilidade de colisão deve ser fornecida ao modelo. Neste contexto, é assumido que não existe uma solução capaz de garantir 0% de falha (colisão). Em (BLACKMORE; ONO; WILLIAMS, 2011), o desvio de obstáculos foi considerado em conjunto com restrições ao risco incorrido. O problema se torna não convexo ao incorporar o desvio de obstáculos, levando a proposição de relaxações e aproximações na tentativa de solucionar o problema.

Em uma outra linha de trabalho, um sistema chamado Kongming é proposto em (LI, 2010) para gerar uma sequência de ações discretas e os controles contínuos ótimos a serem seguidos por um veículo autônomo, durante o cumprimento de uma dada missão. Nesse trabalho, não é considerada a incerteza do ambiente, mas o sequenciamento das tarefas e o planejamento da trajetória são tratados em conjunto. O autor propõe um método que incorpora duas inovações. Primeiro, utiliza uma representação compacta de todos os planejamentos (discreto e contínuo), combinando os pontos fortes de um grafo de planejamento (BLUM; FURST, 1997) em ações discretas e tubos de fluxo (HOFMANN, 2006; HOFMANN; WILLIAMS, 2006) em ações contínuas. Na segunda inovação, propõe uma linguagem formal para projetar missões (MATTMULLER; RINTANEN, 2007) com ações contínuas e discretas em conjunto. O objetivo da missão não é apenas alcançar um estado final num tempo definido, mas alcançar o estado final dentro de um limite inferior e superior de tempo, tornando o horizonte de planejamento dinâmico.

Tomando como base os trabalhos anteriores, (ONO; WILLIAMS; BLACKMORE, 2013) propõem o problema *Chance Constraint Qualitative State Plan* (CCQSP) que une o tratamento da incerteza junto com o modelo formal de uma missão. Torna-se possível realizar o planejamento da missão dentro de uma margem de risco tolerável, solucionando, assim, o sequenciamento das tarefas da missão e o planejamento da rota em conjunto. Novas características foram consideradas no CCQSP, como o tratamento do risco na saturação dos controles do VANTs, permitindo amortecer o crescimento da incerteza do sistema ao longo do tempo. Também foram incorporados outros objetivos como: minimizar o tempo de cumprimento da missão, a distância percorrida e a energia consumida. Os autores propõem um algoritmo planejador, chamado *p-Sulu Planner*, que utiliza *solvers* não lineares para resolver as incertezas e o algoritmo *branch-and-bound* para determinar o sequenciamento ótimo das tarefas.

As mudanças reportadas nos trabalhos citados adicionaram melhorias substanciais ao planejamento de missão, considerando situações mais próximas a um ambiente real e ao cumprimento de tarefas atendendo as diferentes missões. Essas abordagens resol-



veram cenários com veículos aéreos não tripulados (VANTs) (LI, 2010; BLACKMORE; ONO; WILLIAMS, 2011; ONO; WILLIAMS; BLACKMORE, 2013), veículos submarinos autônomos (LI, 2010; LI; WILLIAMS, 2011) e naves espaciais de carga autônoma (ONO; WILLIAMS; BLACKMORE, 2013). Todavia, os modelos se mostraram computacionalmente proibitivos, quando aplicados a cenários mais complexos. Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) mencionam que trabalhos futuros devem propor alternativas que permitam resolver o problema com um grande número de obstáculos.

A Figura 4 ilustra o problema de escalabilidade utilizando 16 cenários diferentes como reportado em (ONO; WILLIAMS; BLACKMORE, 2013). O objetivo consiste em partir do ponto (0, 12) e chegar ao ponto (24, 12). Os quadrados maiores em cinza representam obstáculos e os quadrados menores representam pontos de passagem. Os pontos de passagem estabelecem uma ordem na trajetória e estão dispostos nas coordenadas (9,9), (9, 15), (15, 15) e (15, 9). Em cada cenário, há o tempo em segundos que o método descrito em (ONO; WILLIAMS; BLACKMORE, 2013) levou para planejar a missão.

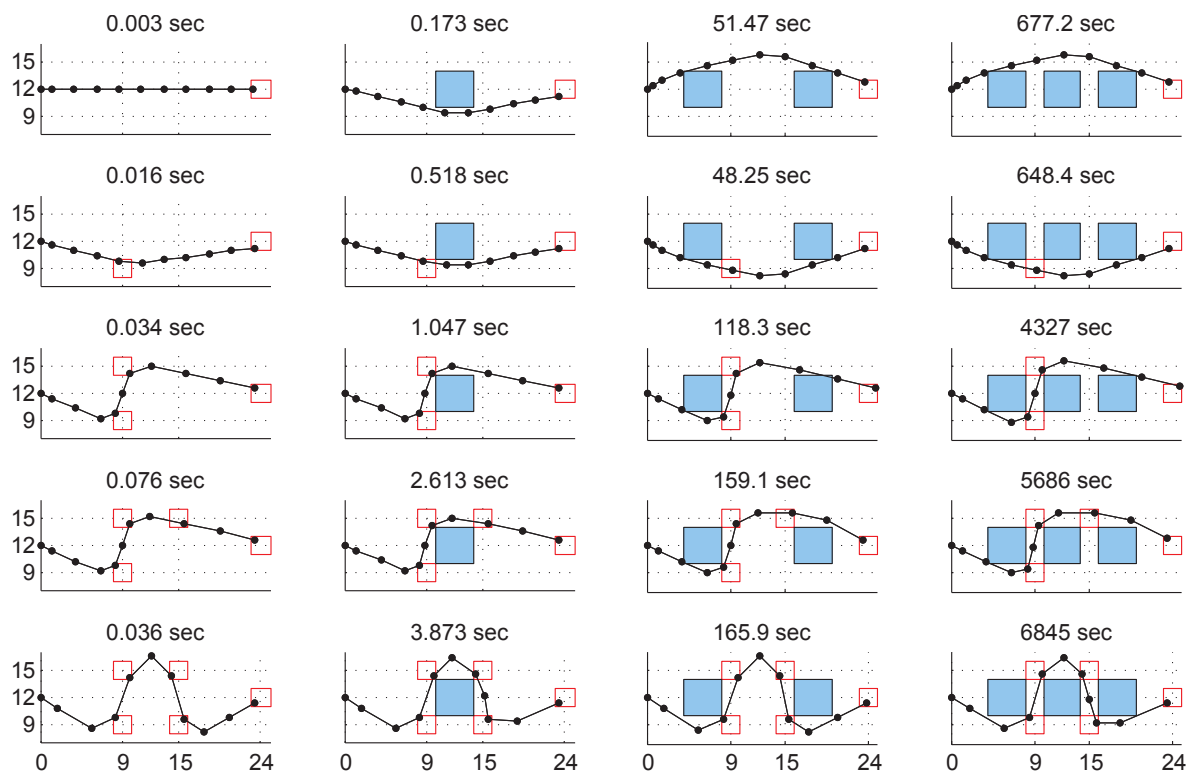


Figura 4 – Problema de escalabilidade: Fonte (ONO; WILLIAMS; BLACKMORE, 2013)

Comparando horizontalmente os cenários, observa-se que o tempo cresce com o aumento do número de obstáculos. Isso é citado em (ONO; WILLIAMS; BLACKMORE, 2013) como um crescimento exponencial. Observando as figuras verticalmente, o aumento do número de tarefas (pontos de passagem) no problema também apresenta um crescimento de tempo considerável, mas em menor escala. Vale ressaltar que os autores não fizeram uma análise do impacto quanto ao aumento do número de *waypoints* sobre o planejamento

da rota. O horizonte de planejamento foi dividido apenas nas 10 unidades de tempo (*waypoints*) como mostrado na Figura 4.

Existem várias técnicas que abordam problemas relacionados na literatura (MAC *et al.*, 2016; PADEN *et al.*, 2016; GONZÁLEZ *et al.*, 2016). Nesses trabalhos, o veículo autônomo é definido como uma partícula no espaço sujeito a campos de força, e utilizam método de campos potenciais (MCP), em que a partícula se move atraída pelos destinos e repelida pelos obstáculos aplicando uma função potencial. As vantagens do método são proporcionar trajetórias dinâmicas que sejam viáveis e suaves. As desvantagens são que o método pode ficar preso em mínimos locais e possui uma descrição imprecisa dos obstáculos, uma vez que eles são descritos por funções contínuas e diferenciáveis (SCHOUWENAARS, 2005). Assim, o MCP pode falhar para cenários com obstáculos próximos uns dos outros (MAC *et al.*, 2016).

As representações utilizadas nesta tese descrevem os obstáculos como um conjunto de hiperplanos e utilizam restrições para o risco de colisão. Uma alternativa para gerar trajetória viável para este tipo de representação é o uso de *Sampling-Based motion Planning* (SBP) como o *Probabilistic Roadmap Method* (PRM) (ALTEROVITZ; SIMEON; GOLDBERG, 2007) e o *Rapidly-exploring Random Tree* (RRT) (LUDERS; KOTHARI; HOW, 2010). O SBP liga os pontos amostrando aleatoriamente o espaço, onde a principal diferença entre PRM e RRT é a maneira de definir o grafo ou a árvore de conexão dos pontos durante a busca (MAC *et al.*, 2016). Esse grafo possui o conjunto de pontos alcançáveis, bem como as transições viáveis (PADEN *et al.*, 2016). As conexões no PRM são definidas offline e os caminhos são estabelecidos online, sendo selecionados aleatoriamente. As conexões e os caminhos no RRT são definidos online, através de uma pesquisa aleatória que estende os ramos para os estados de destino (SCHOUWENAARS, 2005). Os métodos do tipo SBP apresentam a vantagem de lidar melhor com problemas de planejamento complexos e críticos no tempo, no entanto, possuem a desvantagem de retornar soluções sub-ótimas e cair em mínimos locais (GONZÁLEZ *et al.*, 2016).

O objetivo desta tese é obter melhorias nas soluções do problema através de uma formulação baseada em programação linear inteira mista (PLIM), capaz de retornar soluções ótimas. Os autores em (AOUDE *et al.*, 2013) mencionam a falta de ferramentas baseadas em PLIM, que sejam escaláveis com o número de obstáculos e pontos da rota (*waypoints*). Nesse contexto, os mesmos autores argumentam favoravelmente sobre métodos SBP. No entanto, nesta tese entende, considera-se que a escalabilidade em termos de obstáculos e *waypoints* pode ser aprimorada através dos formalismos matemáticos introduzidos aqui. Não é o objetivo deste trabalho dizer que técnicas baseadas em PLIM são uma alternativa melhor do que outras técnicas relevantes. O objetivo é avançar usando uma estrutura baseada em PLIM para planejadores ótimos de missão, incrementando assim os avanços propostos por (DEITS; TEDRAKE, 2015; BLACKMORE; ONO; WILLIAMS, 2011; ONO,

2012; LI, 2010).

Os planejadores de missão desenvolvidos aqui consideram um ambiente geométrico observável e conhecido, mas considerando que estarão sempre presentes as incertezas associadas às observações e ao conhecimento do ambiente. Os autores em (ALTEROVITZ; SIMEON; GOLDBERG, 2007) maximizam a chance de alcançar um objetivo sem colisões apesar da incerteza relacionada ao movimento do robô. *Stochastic Motion Roadmap* (SMR) é combinado com *Markov Decision Process* (MDP), sendo o problema resolvido em tempo polinomial aplicando programação dinâmica. Os autores em (BERG; ABBEEL; GOLDBERG, 2011) combinaram RRT com modelo do tipo *Linear Quadratic Gaussian (LQG) control* para o planejamento de movimento com incerteza. O modelo define, a priori, a distribuição de probabilidade dos estados para avaliar os caminhos e desvio de obstáculos, em que o RRT estabelece os possíveis caminhos. Uma abordagem similar no tratamento da incerteza é descrita em (BRY; ROY, 2011) utilizando *Rapidly-exploring Random Belief Tree* (RRBT). Nesse trabalho os autores mostram que o RRBT consegue retornar uma solução ótima quando infinitas amostras são utilizadas.

O uso de *Partially Observable Markov Decision Processes* (POMDPs) é estudado em (BERG; PATIL; ALTEROVITZ, 2012) e seu uso com *Stochastic Differential Dynamic Programming* (SDDP) é aplicado em (BERG; PATIL; ALTEROVITZ, 2016) e uma abordagem interativa de LQG (iLQG) é introduzida em (BERG; PATIL; ALTEROVITZ, 2012). O *Stochastic Extended Linear Quadratic Regulator* (SELQR) é proposto em (SUN; BERG; ALTEROVITZ, 2016), em que se utiliza uma descrição linear para a estocasticidade de dinâmica e uma descrição quadrática para a função de custo.

Existem algumas diferenças entre a codificação de um PLIM e um LQG. Uma diferença importante está em como é representado o desvio de obstáculos. No PLIM, as incertezas são tratadas com uso de restrições lineares em que o risco de falha tolerável é explicitamente informado pelo usuário. No LQG, parâmetros são ser definidos baseando-se no conhecimento humano para tratar adequadamente o desvio de obstáculos como explicado por (TANG; KUMAR, 2015). Custo computacional e escalabilidade também são um problema a ser enfrentado em um LQG.

Um problema crucial enfrentado nos planejadores de missão está na discretização do horizonte de planejamento. O nível desta discretização pode permitir que os limites de risco entre pontos consecutivos da trajetória sejam excedidos. Por exemplo, nas formulações descritas em (BLACKMORE; ONO; WILLIAMS, 2011), (ONO, 2012) e (ONO; WILLIAMS; BLACKMORE, 2013), os riscos são computados apenas nos pontos da trajetória, sendo ignorados os riscos reais de uma trajetória contínua. A Figura 4 demonstra que existem instantes intermediários entre os pontos da rota em que há interseção com os cantos dos obstáculos. Isso pode ocasionar colisões com probabilidade acima do limite especificado, caso essas rotas sejam de fato executadas.

Nesta tese, foram propostas melhorias para lidar com esse tipo de situação. Existem trabalhos recentes (DEITS; TEDRAKE, 2015; TANG; KUMAR, 2015) que utilizam PLIM para contornar esse tipo de problema no desvio de obstáculos. Os autores em (DEITS; TEDRAKE, 2015) tratam o desvio de obstáculos gerando regiões seguras convexas e sobrepostas para garantir que todo o seguimento de reta está em uma região segura. Uma ferramenta gulosa chamada IRIS foi utilizada para criar um pequeno número de regiões convexas que devem cobrir toda a região livre. O trabalho em (TANG; KUMAR, 2015) apresenta algumas ideias similares às desenvolvidas nesta tese para evitar colisões de um quadricóptero com obstáculos. Contudo, há diferenças significativas entre as contribuições desta tese e as contribuições propostas em (DEITS; TEDRAKE, 2015; TANG; KUMAR, 2015). Primeiro: nesta tese são tratadas as colisões considerando incertezas, em que o estado do veículo não é conhecido com precisão e podem ocorrer distúrbios. Segundo: diferente de (DEITS; TEDRAKE, 2015), a tese propõe tratar os obstáculos diretamente, ou seja, sem a necessidade de nenhuma transformação no cenário. A transformação aplicada por (DEITS; TEDRAKE, 2015) cria um pequeno número de regiões convexas, mas perde parte considerável da região factível do problema, como pode ser visto nos resultados em (DEITS; TEDRAKE, 2015). Terceiro: o quadricóptero com o cabo suspenso considerado em (TANG; KUMAR, 2015) está sempre do mesmo lado de um obstáculo, mas nada é feito a respeito dos pontos consecutivos da trajetória conforme proposto nesta tese.

Na Figura 4 foi visto que a resolução dos problemas apresentados em (BLACKMORE; ONO; WILLIAMS, 2011; ONO; WILLIAMS; BLACKMORE, 2013; LI, 2010), considerando maior quantidade de obstáculos ou maior discretização do horizonte de planejamento, pode se tornar inviável através da solução exata dos modelos propostos naqueles trabalhos. Por outro lado, métodos heurísticos, metaheurísticos ou híbridos ainda não foram aplicados aos problemas mencionados. Métodos heurísticos podem gerar uma boa solução para um problema de otimização, mas sem, necessariamente, comprovar a otimalidade da mesma. Uma metaheurística soluciona uma ampla gama de problemas de otimização, sem ter que se adaptar profundamente ao contexto de cada problema tratado. Desta forma, metaheurísticas são, geralmente, aplicadas a problemas para os quais não há método específico capaz de solucionar o problema de forma satisfatória. Tais métodos têm sido aplicados em problemas reais nas mais diversas áreas (NGUYEN; YANG; BRANKE, 2012).

Os métodos de computação evolutiva são um tipo de metaheurística, baseados em população de soluções, onde soluções são geradas a partir das atuais através de operadores de recombinação e mutação. Trata-se de uma abordagem inspirada na teoria da evolução de Darwin e também aplicada a vários problemas de otimização (BOUSSAÏD; LEPAGNOT; SIARRY, 2013; JIN, 2011). Um dos métodos mais conhecido e utilizado na computação evolutiva é o algoritmo genético (AG), devido, principalmente, a sua flexibilidade e simplicidade. O AG combina, de forma eficiente, a intensificação e exploração durante o

---

processo de busca por soluções, obtendo soluções de boa qualidade dentro de um tempo computacional razoável (GOLDBERG, 1989; MICHALEWICZ, 1996).

Métodos híbridos combinam diferentes algoritmos na expectativa de que suas boas características atuem de forma complementar, melhorando o desempenho do método como um todo. Uma revisão da literatura considerando metaheurísticas híbridas é apresentada em (BLUM *et al.*, 2011). A combinação de métodos também pode incluir um método heurístico ou metaheurístico com técnicas de programação matemática. Os autores em (BALL, 2011) apresentam uma revisão da literatura, considerando heurísticas baseadas em programação matemática.

Um método híbrido é proposto em (TOLEDO *et al.*, 2013b) combinando algoritmo genético (AG) com a resolução exata de submodelos PLIM e com uma heurística baseada em programação matemática do tipo fix and optimize. As variáveis binárias do problema eram definidas como fixas ou livres na representação da solução (indivíduo do AG). Durante a avaliação de uma solução, um submodelo de PLIM era solucionado otimizando um conjunto de variáveis livres, bem como as demais variáveis contínuas do problema. A busca na vizinhança da melhor solução obtida era intensificada pela aplicação da heurística do tipo fix and optimize. As soluções encontradas por esta abordagem superaram a maior parte dos resultados até então reportados na literatura.

Um algoritmo evolutivo híbrido foi utilizado com sucesso em (TOLEDO *et al.*, 2013a), onde um problema real de scheduling foi modelado usando PLIM. Um conjunto considerável de testes foi realizado para avaliar a escalabilidade em várias instâncias do problema real tratado. O método evolutivo foi hibridizado com outra metaheurística, simulated annealing (KIRKPATRICK; GELATT; VECCHI, 1983), e com uma heurística construída especificamente para resolver parte do problema chamada cavity heuristic. Os testes de escalabilidade, usando um método exato para solucionar o modelo PLIM, mostraram que soluções ótimas foram encontradas, quando o modelo continha no máximo 150 variáveis binárias. Soluções factíveis foram encontradas, quando o modelo comportava no máximo 940 variáveis binárias, dentro do tempo limite de uma hora de execução. Por outro lado, o algoritmo evolutivo híbrido conseguia determinar soluções factíveis no mesmo tempo para modelos PLIM com até 8693 variáveis binárias. Apenas para fazer uma comparação, a complexidade do problema do tipo PLIM no trabalho de (ONO; WILLIAMS; BLACKMORE, 2013), e citado na Figura 4, permitiu que soluções ótimas fossem obtidas com no máximo 120 variáveis binárias.

Computação evolutiva também tem sido aplicada em problemas envolvendo planejamento de trajetória em VANTs (HASIRCIOGLU; TOPCUOGLU; ERMIS, 2008; BESADA-PORTAS *et al.*, 2013; EDISON; SHIMA, 2011). Um ambiente 3D é apresentado em (HASIRCIOGLU; TOPCUOGLU; ERMIS, 2008), cujo desafio, era encontrar uma trajetória capaz de evitar a colisão com as ondulações no terreno. Assim um algoritmo evolutivo

foi proposto para determinar uma trajetória contínua para a aeronave, considerando as ondulações existentes. Em outro trabalho, a otimização de objetivos conflitantes é estudada, ([BESADA-PORTAS \*et al.\*, 2013](#)), em que o desempenho de vários métodos, baseados em algoritmos genéticos, enxame de partículas e evolução diferencial, são avaliados para o planejamento de trajetória em VANTs. O planejamento de trajetória para vários VANTs em voo simultâneo é tratado em ([EDISON; SHIMA, 2011](#)) usando algoritmo genético.

## FUNDAMENTAÇÃO MATEMÁTICA

Os modelos para os planejadores propostos precisam ser definidas com o uso de restrições lineares para viabilizar o uso de PLIM. Assim, esta seção reúne e organiza os conhecimentos e estratégias de diversas fontes para construir um modelo formal de como transformar lógicas comuns em restrições puramente lineares.

### 3.1 Conectivos lógicos

Nesta seção será visto como modelar conectivos lógicos envolvendo proposições (variáveis binárias). Para isso, iniciaremos com a construção de uma tabela verdade de restrições, conforme apresentado na Tabela 1. Dado duas variáveis lógicas,  $P_1$  e  $P_2$  quaisquer, podemos definir que uma nova variável lógica  $y$  respeite qualquer tabela verdade de duas proposições ao codificar as restrições para  $y = 1$  e  $y = 0$  de maneira apropriada.

Propositions		Constraints		Complexity		
$P_1$	$P_2$	for $y = 1$	for $y = 0$	$B$	$C$	$R$
0	0	$y \geq 1 - P_1 - P_2$	$y \leq P_1 + P_2$	-	1	1
0	1	$y \geq P_2 - P_1$	$y \leq 1 + P_1 - P_2$	-	1	1
1	0	$y \geq P_1 - P_2$	$y \leq 1 - P_1 + P_2$	-	1	1
1	1	$y \geq P_1 + P_2 - 1$	$y \leq 2 - P_1 - P_2$	-	1	1

Tabela 1 – Tabela verdade ( $Tv$ )

As duas primeiras colunas da Tabela 1 apresentam todas as combinações para as proposições  $P_1$  e  $P_2$ , as terceira e quarta colunas apresentam as restrições que ligam  $y$  às proposições seguindo a seguinte regra: caso desejemos que a resposta seja  $y = 1$  implementa-se a restrição da terceira coluna, caso deseje-se  $y = 0$  implementa-se conforme a quarta coluna. Assim, é possível utilizar a Tabela 1 para se construir qualquer lógica envolvendo duas proposições. Por exemplo, caso desejemos definir que uma variável  $y$  seja equivalente a  $P_1 \wedge P_2$ , precisaremos adicionar as restrições das 3 primeiras linhas da



coluna  $y = 0$  e a restrição na última linha da coluna  $y = 1$ . Para simplificar essa ideia para os próximos exemplos, vamos denotar  $Tv = [0 \ 0 \ 0 \ 1]$  para indicar as restrições a serem adicionadas para o conectivo  $P_1 \wedge P_2$  descrito acima. Nesta notação, o conectivo  $P_1 \vee P_2$  seria  $Tv = [0 \ 1 \ 1 \ 1]$ .

Com este formalismo introduzido, temos uma maneira sistemática para conseguir representar qualquer tabela verdade de duas proposições. Um fato a ser notado é que se  $P_1$  e  $P_2$  são variáveis binárias, então  $y \in \mathbb{R}^+$  também assumirá valor 0 ou 1. Isto significa que trabalhar com conectivos lógicos entre duas proposições não exige a definição de novas variáveis binárias. As três últimas colunas da Tabela 1 apresentam este resultado, indicando quantas variáveis binárias ( $B$ ), variáveis contínuas ( $C$ ) e restrições ( $R$ ) a mais serão necessárias ao elaborar cada restrição de um conectivo. Até aqui, sabemos que não precisaremos de novas variáveis binárias (-), entretanto devemos nos perguntar se esta é a forma mais eficiente de fazê-lo. Por exemplo, ao utilizar as restrições  $Tv = [0 \ 0 \ 0 \ 1]$  para o conectivo  $P_1 \wedge P_2$  precisaremos de uma nova variável contínua  $y$  e de 4 restrições. Entretanto, é possível notar que:

$$y \leq P_1 \subset \left\{ \begin{array}{l} y \leq P_1 + P_2 \\ y \leq 1 + P_1 - P_2 \end{array} \right\} \quad (3.1)$$

$$y \leq P_2 \subset \left\{ \begin{array}{l} y \leq P_1 + P_2 \\ y \leq 1 - P_1 + P_2 \end{array} \right\} \quad (3.2)$$

Uma forma mais fechada da região factível pode ser elaborada para  $y$ ,  $P_1$  e  $P_2$  de forma que, para o conectivo  $P_1 \wedge P_2$ , teremos:

$$\left\{ \begin{array}{l} y \leq P_1 \\ y \leq P_2 \\ y \geq P_1 + P_2 - 1 \end{array} \right\} \subset \left\{ \begin{array}{l} y \leq P_1 + P_2 \\ y \leq 1 + P_1 - P_2 \\ y \leq 1 - P_1 + P_2 \\ y \geq P_1 + P_2 - 1 \end{array} \right\} \quad (3.3)$$

Para o conectivo,  $\wedge$  exemplificado, precisaremos apenas de 3 restrições e teremos uma região factível ainda menor no espaço de busca. É possível provar que não existe uma solução melhor do que a apresentada acima, uma vez que as 3 restrições descritas formam a envoltória convexa de  $y = P_1 \wedge P_2$ . Logo, os pontos  $(P_1, P_2, y) = \{(0,0,0); (0,1,0); (1,0,0); (1,1,1)\}$  são os quatro vértices do tetraedro formado pelas 3 novas restrições e o limitante inferior  $y \geq 0$ , como apresentado na Figura 5. Ao conseguirmos desenhar a envoltória convexa do conectivo  $\wedge$ , prova-se que não existe outra forma melhor de fazê-lo, uma vez que a envoltória convexa é a menor região convexa, que preserva todas as combinações inteiras do conectivo  $\wedge$ .



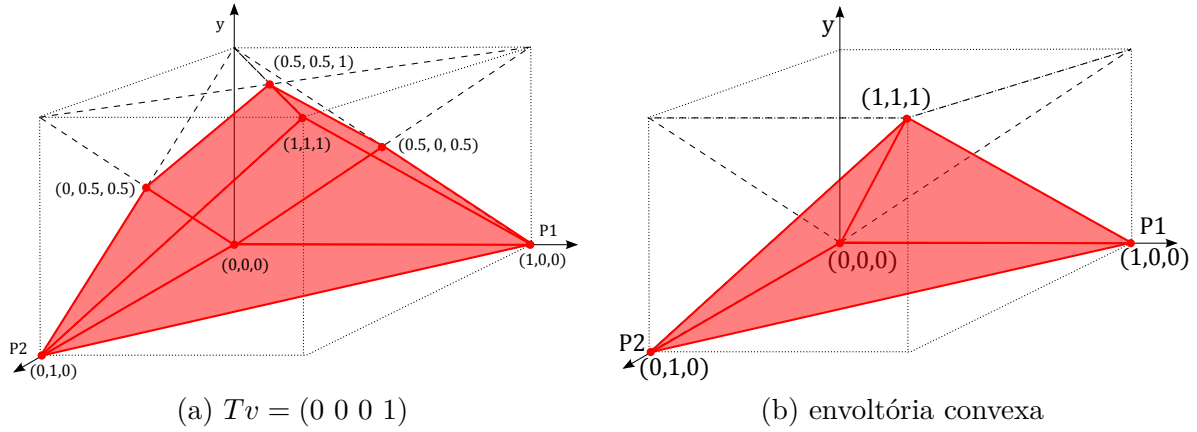


Figura 5 – Exemplo com duas regiões diferentes que modelam o conectivo  $y = P_1 \wedge P_2$

$[0\ 0\ *\ *]$	$[0\ *\ 0\ *]$	$[*\ *\ 0\ 0]$	$[*\ 0\ *\ 0]$
$y \leq P_1$	$y \leq P_2$	$y \leq 1 - P_1$	$y \leq 1 - P_2$
$[*\ *\ 1\ 1]$	$[*\ 1\ *\ 1]$	$[1\ 1\ *\ *]$	$[1\ *\ 1\ *]$
$y \geq P_1$	$y \geq P_2$	$y \geq 1 - P_1$	$y \geq 1 - P_2$

Tabela 2 – Otimizando a tabela verdade ( $Tv$ )

Baseado nestes conceitos, a Tabela 2 apresenta uma regra geral para construir a envoltória convexa de qualquer tabela verdade com duas proposições. Como exemplo, voltamos ao conectivo  $\wedge$  onde temos a tabela verdade  $Tv = [0\ 0\ 0\ 1]$ . Aplicando as regras, podemos tomar  $Tv = [0\ 0\ *\ *] \Leftrightarrow y \leq P_1$  e  $Tv = [0\ *\ 0\ *] \Leftrightarrow y \leq P_2$ , isto resulta em um padrão  $Tv = [0\ 0\ 0\ *]$ . Como não é possível utilizar mais nenhuma regra da Tabela 2, deve-se utilizar restrição original da Tabela 1 para qualquer  $*$  que sobrar ( $y \geq P_1 + P_2 - 1$ ) e teremos  $Tv = [0\ 0\ 0\ 1]$ . É possível provar que, ao utilizar as regras da Tabela 2 para otimizar a Tabela 1 definida anteriormente, o resultado sempre será uma envoltória convexa para qualquer tabela verdade  $Tv$ , exceto nas duas situações seguintes:  $Tv = [0\ 0\ 0\ 0]$  e  $Tv = [1\ 1\ 1\ 1]$ . Entretanto, estes dois casos tem uma codificação trivial da tabela verdade, pois implicam que  $y = 0$  e  $y = 1$  para cada uma das duas situações, respectivamente. A prova de que a aplicação das regras da Tabela 2 resulta sempre em uma envoltória convexa é obtida por inspeção, ao verificar que para todas as 14 combinações restantes de  $Tv$ , os vértices da região definida pelas restrições selecionadas estão nos quatro pontos inteiros de  $(P_1, P_2, y)$ .

**Lema 1.** Utilizar as regras da Tabela 2 para otimizar a Tabela 1 sempre resultará numa envoltória convexa para qualquer  $Tv$  tal que  $Tv \neq [0\ 0\ 0\ 0]$  e  $Tv \neq [1\ 1\ 1\ 1]$ .

*Demonstração.* A Figura 6 apresenta a aplicação das regras da Tabela 2 para todas as combinações possíveis da Tabela 1, onde é possível observar que todas as restrições estão na envoltória convexa, exceto para  $Tv = [0\ 0\ 0\ 0]$  e  $Tv = [1\ 1\ 1\ 1]$ , cuja envoltória convexa é formada por  $y = 0$  e  $y = 1$ , respectivamente.

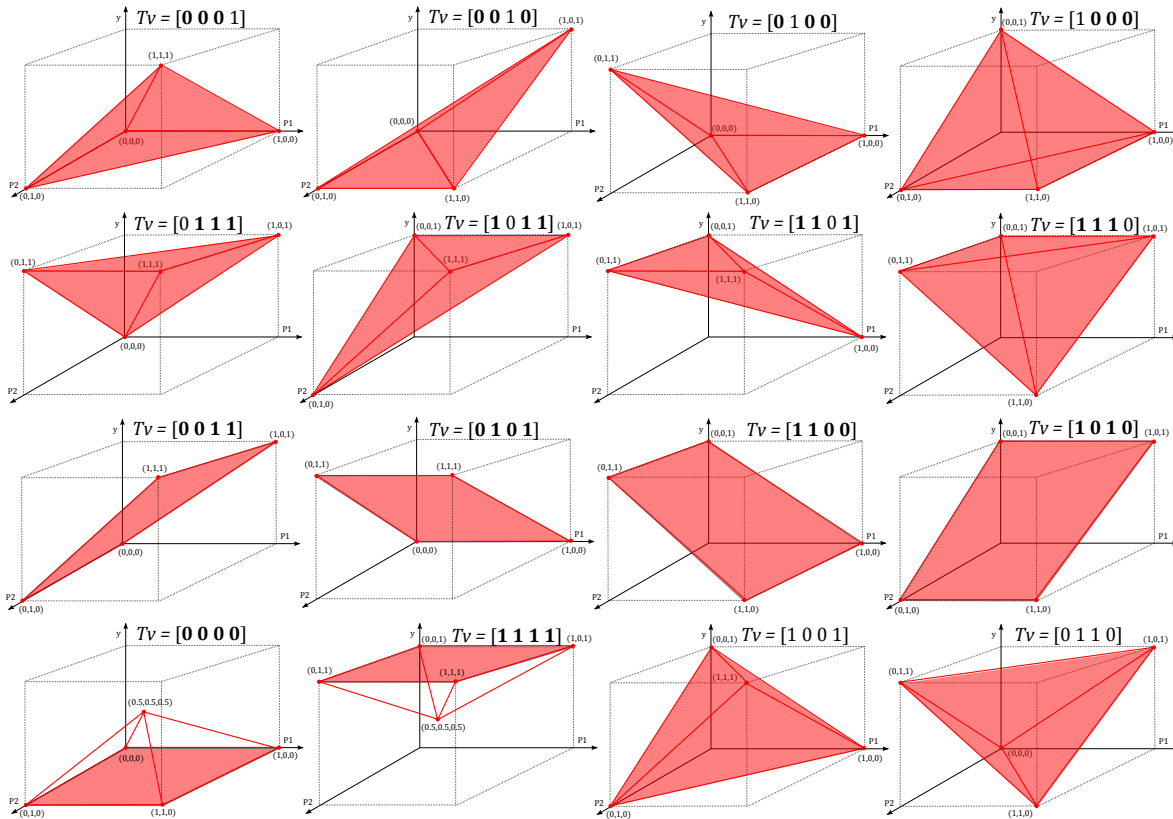


Figura 6 – envoltória convexa de todas as tabelas verdade de duas proposições

□

Uma codificação eficiente pode ser definida com esta técnica para os conectivos lógicos mais comuns. Alguns são apresentados na Tabela 3. As últimas 3 colunas desta tabela mostram que nenhuma variável binária extra precisará ser adicionada. Apenas uma nova variável  $y$  contínua precisa ser definida para armazenar o resultado do conectivo e o número de restrições pode variar um pouco dependendo do conectivo. Os valores em negrito dos padrões na tabela verdade  $Tv$  indicam as restrições que foram otimizadas. Nota-se que não existem regras capazes de otimizar  $y = P_1 \bar{\vee} P_2$  e  $y = P_1 \leftrightarrow P_2$ , isto porque as regras originais compostas pelas 4 restrições padrão já formam a envoltória convexa destes dois conectivos, conforme apresentado no Lema 1.

Nome	Notação Lógica	Interpretação	Restrições	$B$	$C$	$R$
$L1$	$y = \neg P_1$	$y = \text{not } P_1$	$y \text{ is } 1 - P_1$	-	-	-
$L2$	$y = P_1 \wedge P_2$	$y = P_1 \text{ and } P_2$	$Tv[\mathbf{0} \ \mathbf{0} \ \mathbf{0} \ 1]$	-	1	3
$L3$	$y = P_1 \vee P_2$	$y = P_1 \text{ or } P_2$	$Tv[\mathbf{0} \ \mathbf{1} \ \mathbf{1} \ 1]$	-	1	3
$L4$	$y = P_1 \bar{\vee} P_2$	$y = P_1 \text{ xor } P_2$	$Tv[\mathbf{0} \ \mathbf{1} \ \mathbf{1} \ 0]$	-	1	4
$L5$	$y = P_1 \rightarrow P_2$	$y = \text{if } P_1 \text{ then } P_2$	$Tv[\mathbf{1} \ \mathbf{1} \ \mathbf{0} \ 1]$	-	1	3
$L6$	$y = P_1 \leftrightarrow P_2$	$y = P_1 \text{ if and only if } P_2$	$Tv[\mathbf{1} \ \mathbf{0} \ \mathbf{0} \ 1]$	-	1	4

Tabela 3 – Conectivos básicos

A seguir, a generalização de alguns conectivos lógicos é apresentada para uma situação em que temos um conjunto de proposições:  $P_i \mid i = 1 \dots n$ . Conforme apresentado

na Tabela 4, a generalização dos conectivos  $\wedge$  e  $\vee$  precisará cada uma de uma variável  $y$  contínua e  $n + 1$  restrições.

Nome	Notação Lógica	Restrições	Usada	$B$	$C$	$R$
$G1$	$y = \bigwedge_i P_i$	$y \geq 1 - n + \sum_i P_i$ $\forall_i [y \leq P_i]$		-	1	$n + 1$
$G2$	$y = \bigvee_i P_i$	$y \leq \sum_i P_i$ $\forall_i [y \geq P_i]$		-	1	$n + 1$
$G3$	$y = \bar{\bigvee}_i P_i$	$y = p_n \bar{\bigvee} z_{n-1}$ $z_i = p_i \bar{\bigvee} z_{i-1}$ $z_2 = p_2 \bar{\bigvee} p_1$	(L4) (L4) (L4)	-	$n - 1$	$4n - 4$
$G4$	$y = \bigwedge_i P_i \rightarrow \bigvee_j P_j$ $\forall i \neq \forall j$	$z = \bigwedge_i P_i$ $w = \bigvee_j P_j$ $y = z \rightarrow w$	(G1) (G2) (L5)	-	3	$n + m + 5$
$G5$	$y = \bigvee_i P_i \rightarrow \bigwedge_j P_j$ $\forall i \neq \forall j$	$z = \bigvee_i P_i$ $w = \bigwedge_j P_j$ $y = z \rightarrow w$	(G2) (G1) (L5)	-	3	$n + m + 5$
$G6$	$y = \bigwedge_i P_i \leftrightarrow \bigvee_j P_j$ $\forall i \neq \forall j$	$z = \bigwedge_i P_i$ $w = \bigvee_j P_j$ $y = z \leftrightarrow w$	(G1) (G2) (L6)	-	3	$n + m + 6$

Tabela 4 – Generalização dos conectivos básicos

**Teorema 1.** Os conectivos  $\wedge$  e  $\vee$  codificados conforme apresentado na Tabela 4 também formam uma envoltória convexa no espaço  $\mathbb{R}^{n+1}$ .

*Demonstração.* **Provando para o conectivo  $\wedge$ :**

1. Sabe-se que  $P = [0, 1]^n$  é uma envoltória convexa para todas as combinações inteiras das proposições de  $P_i \mid i = 1 \dots n$ . Isso é verdade, já que qualquer combinação inteira das proposições de  $P_i$  está nos vértices da região  $P$ .
2. Para  $y = 0$ , teremos que  $\sum_i P_i \leq n - 1$ , dada a primeira restrição do conectivo  $\wedge$ . Assim podemos definir uma região  $P' \subset P$ , onde  $P' = \{P \mid \sum_i P_i \leq n - 1\}$ . A região  $P'$  também é uma envoltória convexa, visto que a restrição define um hiperplano que exclui o vértice  $v = \{1, \dots, 1\} \in P$ , mas tem intercessão com todos os outros  $n$  vértices  $v'$  de  $P$  nos quais se ligavam diretamente a  $v$  pela distância de uma unidade:  $|v - v'| = 1$ .
3. Para  $y = 1$ , teremos que  $\forall_i [P_i = 1]$ , dada a segunda restrição do conectivo  $\wedge$ , definindo assim um novo ponto  $v'' \in \mathbb{R}^{n+1}$  que estabelece  $y = 1$  exatamente para a combinação de  $P_i$  excluída de  $P$ .
4. Conclusão, ao unir os itens 2 e 3 em uma nova região  $C$  no espaço  $\mathbb{R}^{n+1}$ , composta pelos vértices da envoltória convexa  $P' \in \mathbb{R}^n$  para  $y = 0$ , mais um novo vértice  $v'' = \{y = 1, P_i = 1\}$  para  $y = 1$ , estamos definindo uma nova envoltória convexa, concluindo a demonstração.

□

*Demonstração.* **Provando para o conectivo  $\vee$ :** A prova para o conectivo  $\vee$  segue por demonstração direta, visto que esse conectivo é o complemento do conectivo  $\wedge$  que foi provado anteriormente. Assim pelo teorema de De Morgan temos que:

$$\bigwedge_i P_i = \neg \bigvee_i \neg P_i \quad (3.4)$$

Sabemos pela prova anterior que  $y = \bigwedge_i P_i$  pode ser codificado na casca convexa ao utilizar as restrições apresentadas em  $G1$ . Agora, é possível codificar  $y' = \bigvee_i P'_i$  através de De Morgan como um conectivo  $\wedge$  fazendo com que  $y' = \neg y = 1 - y$  e  $P'_i = \neg P_i = 1 - P_i$ . Assim, isolando e substituindo  $y$  e  $P_i$  nas equações em  $G1$ , teremos por dedução direta as equações em  $G2$ , conforme apresentado na equação 3.5 abaixo.

$$\left\{ \begin{array}{l} y \geq 1 - n + \sum_i P_i \\ \forall_i [y \leq P_i] \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} 1 - y' \geq 1 - n + \sum_i (1 - P'_i) \\ \forall_i [1 - y' \leq 1 - P'_i] \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} y' \leq \sum_i P'_i \\ \forall_i [y' \geq P'_i] \end{array} \right\} \quad (3.5)$$

□

Da mesma forma, também é possível codificar vários tipos de restrições lógicas como restrições lineares. Com base nos conceitos apresentados anteriormente chegou-se na Tabela 5 que apresenta restrições lineares uteis para codificar vários tipos de restrições lógicas de uma forma eficiente.

## 3.2 Modelando implicações tautológicas

Vamos utilizar o Teorema 2 para codificar implicações tautológicas ( $\Rightarrow$ ) de variáveis lógicas em conjunto com restrições lineares. Podemos, por exemplo, estabelecer que um conjunto de restrições estarão ativas, caso uma proposição seja verdadeira. O Teorema 2 pode ser encontrado em (SIERKSMA, 1996) e afirma que:

**Teorema 2.** Seja  $f(x) : D \rightarrow \mathbb{R}$  e  $\delta \in \{0, 1\}$ . Seja também  $M$  um valor real não nulo tal que  $M \geq \max \{f(x) \mid x \in D\}$ . Então, para cada  $x \in D$  as seguintes afirmativas são equivalentes:

$$\delta = 0 \quad \Rightarrow \quad f(x) \leq 0 \quad (3.6)$$

$$f(x) - M\delta \leq 0 \quad (3.7)$$

*Demonstração.* Se  $\delta = 0$ , teremos diretamente pela equação 3.7 que  $f(x) \leq 0$ . Se  $\delta = 1$ , temos  $f(x) \leq M$  que resulta em verdade sempre para  $\forall x \in D$ , dada a definição do  $M$ . □

Nome	Notação Lógica	Restrições	Utiliza	B	C	R
R1	$P_1$	fixar $P_1 = 1$		-	-	-
R2	$\neg P_1$	fixar $P_1 = 0$		-	-	-
R3	$P_1 \wedge P_2$	fixar $P_1 = 1$ e $P_2 = 1$	(L2)	-	-	-
R4	$P_1 \vee P_2$	$P_1 + P_2 \geq 1$	(L3)	-	-	1
R5	$P_1 \bar{\vee} P_2$	$P_1 + P_2 = 1$	(L4)	-	-	1
R6	$P_1 \Rightarrow P_2$	$P_1 \leq P_2$	(L5)	-	-	1
R7	$P_1 \Leftrightarrow P_2$	$P_1 = P_2$	(L6)	-	-	-
R8	$\bigwedge_i P_i$	$\forall_i [\text{fixar } P_i = 1]$	(G1)	-	-	-
R9	$\bigvee_i P_i$	$\sum_i P_i \geq 1$	(G2)	-	-	1
R10	$\bar{\bigvee}_i P_i$	$\sum_i P_i = 1$	(R9)	-	-	1
R11	$\bigwedge_i P_i \Rightarrow \bigvee_j P_j$	$1 - n + \sum_i P_i \leq \sum_j P_j$	(G4)	-	-	1
R12	$\bigvee_i P_i \Rightarrow \bigwedge_j P_j$	$\forall(i, j)[P_i \leq P_j]$	(G5)	-	-	$n * m$
R13	$\bigvee_i P_i \Leftrightarrow \bigwedge_j P_j$	$1 - m + \sum_j P_j \leq \sum_i P_i$ $\forall(i, j)[P_i \leq P_j]$	(R11) (R12)	-	-	$n * m + 1$
F1	$y \Rightarrow \neg P_1$	$y \leq 1 - P_1$	(L1)	-	-	1
F2	$y \Rightarrow P_1 \wedge P_2$	$y \leq P_1, y \leq P_2$	(L2)	-	-	2
F3	$y \Rightarrow P_1 \vee P_2$	$y \leq P_1 + P_2$	(L3)	-	-	1
F4	$y \Rightarrow P_1 \bar{\vee} P_2$	$y \leq P_1 + P_2 \leq 2 - y$	(L4)	-	-	2
F5	$y \Rightarrow [P_1 \Rightarrow P_2]$	$y \leq 1 - P_1 + P_2$	(L5)	-	-	1
F6	$y \Rightarrow [P_1 \Leftrightarrow P_2]$	$y - 1 \leq P_1 - P_2 \leq 1 - y$	(L6)	-	-	2
F7	$y \Rightarrow [P_1 \Leftrightarrow \bar{v}]$	Se $\bar{v} = 1$ : $y \leq P_1$ Se $\bar{v} = 0$ : $y \leq 1 - P_1$	(F6)	-	-	1
F8	$y \Rightarrow \bigwedge_i P_i$	$\forall i[y \leq P_i]$	(R12)	-	-	$n$
F9	$y \Rightarrow \bigvee_i P_i$	$y \leq \sum_i P_i$	(R11)	-	-	1
F10	$y \Rightarrow \bar{\bigvee}_i P_i$	$y \leq \sum_i P_i \leq n - y(n - 1)$	(F9)	1	-	2
B1	$\neg P_1 \Rightarrow y$	$y \geq 1 - P_1$	(L1)	-	-	1
B2	$P_1 \wedge P_2 \Rightarrow y$	$y \geq P_1 + P_2 - 1$	(L2)	-	-	1
B3	$P_1 \vee P_2 \Rightarrow y$	$y \geq P_1, y \geq P_2$	(L3)	-	-	2
B4	$P_1 \bar{\vee} P_2 \Rightarrow y$	$y \geq P_2 - P_1, y \geq P_1 - P_2$	(L4)	-	-	2
B5	$[P_1 \Rightarrow P_2] \Rightarrow y$	$y \geq 1 - P_1, y \geq P_2$	(L5)	-	-	2
B6	$[P_1 \Leftrightarrow P_2] \Rightarrow y$	$y + 1 \geq P_1 + P_2 \geq 1 - y$	(L6)	-	-	2
B7	$[P_1 \Leftrightarrow \bar{v}] \Rightarrow y$	Se $\bar{v} = 1$ : $y \geq P_1$ Se $\bar{v} = 0$ : $y \geq 1 - P_1$	(B6)	-	-	1
B8	$\bigwedge_i P_i \Rightarrow y$	$y \geq 1 - n + \sum_i P_i$	(R11)	-	-	1
B9	$\bigvee_i P_i \Rightarrow y$	$\forall i[y \geq P_i]$	(R12)	-	-	$n$
B10	$\bar{\bigvee}_i P_i \Rightarrow y$	$\forall i[y \geq P_i + \sum_{j \neq i} P_j]$	(B9)	-	-	$n$

Tabela 5 – Restrições básicas

A equação 3.7 é amplamente utilizada em modelos de programação linear inteira mista (PLIM). Em problemas de *lotsizing* e *scheduling*, por exemplo, é utilizada para proibir a produção de um item caso não ocorra o setup de máquina. A partir deste ponto, vamos fazer algumas deduções a partir do Teorema 2 para codificar lógicas semelhantes.

**Corolário 1.** Considerando  $y = \neg\delta = 1 - \delta$ , podemos inferir com base no Teorema 2 que as afirmações abaixo também são equivalentes:

$$y \Rightarrow f(x) \leq 0 \quad (3.8)$$

$$f(x) \leq M(1 - y) \quad (3.9)$$

Veja que  $y$  foi utilizado com o significado de  $y = 1$  por simplicidade. Assim, ambas

as equações 3.8 e 3.9 possuem o seguinte significado: se a proposição  $y$  é verdade, então  $f(x) \leq 0$ .

Nome	Notação Lógica	Restrições	Utiliza	$B$	$C$	$R$
$S1$	$y \Rightarrow a_i x \leq b_i$	$a_i x - b_i \leq M(1 - y)$	eq.(3.9)	1	-	1
$S2$	$y \Rightarrow a_i x \geq b_i$	$a_i x - b_i \geq M(y - 1)$	eq.(3.9)	1	-	1
$S3$	$y \Rightarrow a_i x = b_i$	$a_i x - b_i \leq M(1 - y)$ $a_i x - b_i \geq M(y - 1)$	(S1) (S2)	1	-	2
$S4$	$y \Rightarrow a_i x \neq b_i$	$z \Rightarrow a_i x \leq b_i - \epsilon$ $w \Rightarrow a_i x \geq b_i + \epsilon$ $y = z \vee w$	(S1) (S2) (L4)	2	1	5
$S5$	$a_i x \leq b_i \Rightarrow y$	$a_i x - b_i \geq -My + \epsilon$	eq.(3.11)	1	-	1
$S6$	$a_i x \geq b_i \Rightarrow y$	$a_i x - b_i \leq My - \epsilon$	eq.(3.11)	1	-	1
$S7$	$a_i x = b_i \Rightarrow y$	$a_i x \leq b_i \Rightarrow z$ $a_i x \geq b_i \Rightarrow w$ $y = z \wedge w$	(S5) (S6) (L2)	2	1	5
$S8$	$a_i x \neq b_i \Rightarrow y$	$a_i x \leq b_i - \epsilon \Rightarrow z$ $a_i x \geq b_i + \epsilon \Rightarrow w$ $y = z \vee w$	(S5) (S6) (L4)	2	1	5
$S9$	$exp_i \leq q \leq exp_j \Rightarrow y$	$exp_i \leq q \Rightarrow z$ $exp_j \geq q \Rightarrow w$ $y = z \wedge w$	(S5) (S6) (L2)	2	1	5
$S10$	$lb_i \leq exp_i \leq ub_i \Rightarrow y$	$exp_i \leq ub_i \Rightarrow z$ $exp_j \geq lb_i \Rightarrow w$ $y = z \wedge w$	(S5) (S6) (L2)	2	1	5

Tabela 6 – Codificando implicações lógicas para uma simples restrição  $i$  genérica.

Com base no Corolário 1, a Tabela 6 apresenta como codificar lógicas em que uma única restrição  $i$  deve ser obedecida, se uma proposição  $y$  é verdadeira. Na segunda coluna, é apresentada a notação lógica e na terceira como a restrição lógica é codificada. As quatro primeiras linhas  $S1-S4$  apresentam como codificar cada um dos operadores de comparação:  $\leq$ ,  $\geq$ ,  $=$  e  $\neq$ , respectivamente. As três últimas colunas da Tabela 6 apresentam quantas variáveis binárias ( $B$ ), contínuas ( $C$ ) e restrições ( $R$ ) extras serão necessárias ao elaborar cada estrutura lógica.

Apresentaremos agora o Teorema 3 e o Corolário 2, que foram utilizados para montar as lógicas de  $S5-S8$  na Tabela 6.

**Teorema 3.** Seja  $f(x) : D \rightarrow \mathbb{R}$  e  $y \in \{0, 1\}$ . Seja  $\epsilon$  um valor real positivo não nulo, mas próximo de zero, e seja  $M$  um valor real positivo não nulo tal que  $M \geq \max \{-f(x) + \epsilon \mid x \in D\}$ . Então, para cada  $x \in D$  as seguintes afirmativas são equivalentes:

$$f(x) < \epsilon \quad \Rightarrow \quad y \quad (3.10)$$

$$f(x) \geq -My + \epsilon \quad (3.11)$$

*Demonstração.* Se  $f(x) < \epsilon$ , então  $y = 1$ , pois pela equação 3.11 teremos que  $y \geq \frac{-f(x)+\epsilon}{M} > 0$ . Note que  $\frac{-f(x)+\epsilon}{M}$  pode assumir valor máximo de 1 pela definição do  $M$  e, assim, o

domínio de  $f(x)$  continua preservado. Se  $f(x) \geq \epsilon$ , então temos que  $y \geq 0 \geq \frac{-f(x)+\epsilon}{M}$ , note que isso é verdade sempre e independe de  $y$  que ficará livre para valer 0 ou 1.  $\square$

A expressão ( $< \epsilon$ ) utilizada na equação 3.10 se torna um inconveniente para o uso prático do Teorema 3. Para trabalharmos com restrições de uma forma mais cômoda, segue o Corolário 2.

**Corolário 2.** Considerando que  $\epsilon > 0$ , torna-se possível estabelecer:  $f(x) - \epsilon < 0 \cong f(x) \leq 0$  para  $\epsilon \approx 0$  e assim reescrever a equação 3.10 conforme a equação 3.12.

$$f(x) \leq 0 \quad \Rightarrow \quad y \quad (3.12)$$

Por fim, a Tabela 7 apresenta a generalização das lógicas estabelecidas na Tabela 6 para  $n$  restrições funcionando em conjunto.

Nome	Notação Lógica	Restrições	Utiliza	$B$	$C$	$R$
$C1$	$y \Rightarrow Ax \leq b$	$\forall_i [y \Rightarrow a_i x \leq b_i]$	(S1)	1	-	$n$
$C2$	$y \Rightarrow Ax \geq b$	$\forall_i [y \Rightarrow a_i x \geq b_i]$	(S2)	1	-	$n$
$C3$	$y \Rightarrow Ax = b$	$\forall_i [y \Rightarrow a_i x = b_i]$	(S3)	1	-	$2n$
$C4$	$y \Rightarrow Ax \neq b$	$\forall_i [z_i \Rightarrow a_i x \neq b_i]$ $y = \bigvee_i z_i$	(S4) (G2)	$2n$	$n + 1$	$6n + 1$
$C5$	$Ax \leq b \Rightarrow y$	$\forall_i [a_i x \leq b_i \Rightarrow z_i]$ $y = \bigwedge_i z_i$	(S5) (G1)	$n$	1	$2n + 1$
$C6$	$Ax \geq b \Rightarrow y$	$\forall_i [a_i x \geq b_i \Rightarrow z_i]$ $y = \bigwedge_i z_i$	(S6) (G1)	$n$	1	$2n + 1$
$C7$	$Ax = b \Rightarrow y$	$\forall_i [a_i x = b_i \Rightarrow z_i]$ $y = \bigwedge_i z_i$	(S7) (G1)	$2n$	$n + 1$	$6n + 1$
$C8$	$Ax \neq b \Rightarrow y$	$\forall_i [a_i x \neq b_i \Rightarrow z_i]$ $y = \bigvee_i z_i$	(S8) (G2)	$2n$	$n + 1$	$6n + 1$

Tabela 7 – Codificando implicações lógicas para um conjunto com  $n$  restrições genéricas.





# CHANCE-CONSTRAINED PATH-PLANNING PROBLEM

## 4.1 Introdução

Os autores em (BLACKMORE; ONO; WILLIAMS, 2011) introduzem o *Chance-Constrained Path-Planning* (CCPP) como um problema de otimização não convexo e estocástico. A Figura 7 apresenta um exemplo de missão para o CCPP e uma possível solução. Tal missão é descrita informalmente a seguir:

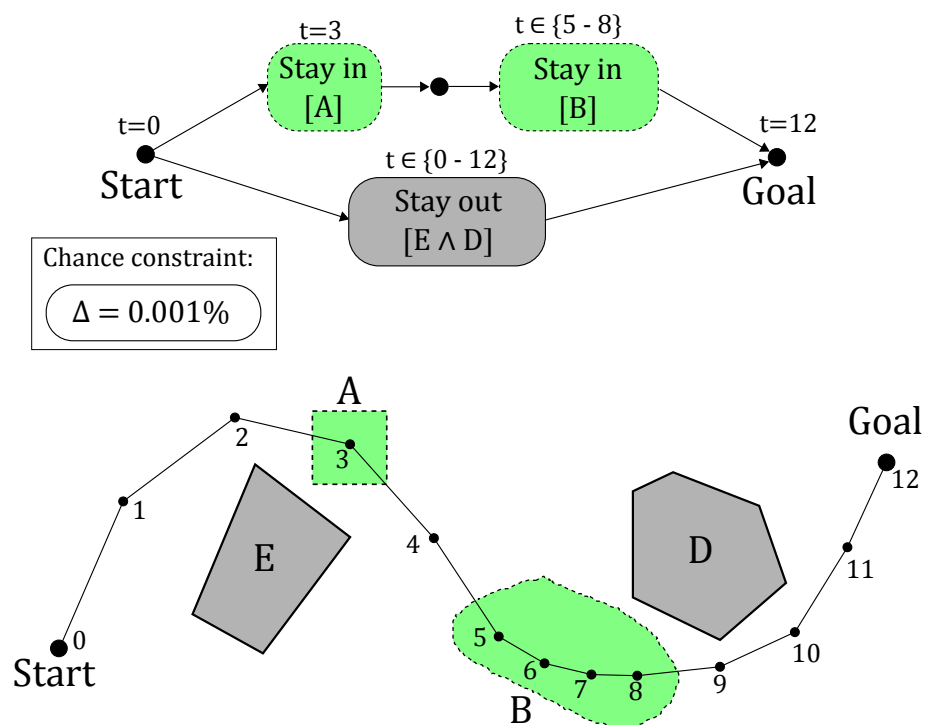


Figura 7 – Um exemplo de missão para o CCPP e sua solução

Iniciando do ponto **Start** no instante  $t = 0$ , o VANT deve alcançar o objetivo final no ponto **Goal** no instante  $t = 12$ , devendo passar pela região **A** no instante  $t = 3$  e permanecer na região **B** nos instantes  $t = 5, 6, 7, 8$ . Durante todo o voo, o VANT deve manter-se na região segura, evitando colisão com os obstáculos **D** e **E**. A probabilidade do VANT deixar as regiões de passagem ou colidir com os obstáculos deve ser de no máximo  $\Delta = 0.001\%$ .

## 4.2 Descrição do problema

Uma formulação matemática será apresentada para o CCPP, seguindo a definição original proposta em (BLACKMORE; ONO; WILLIAMS, 2011).

$$\text{Minimizar } \mathcal{J}(\cdot) \quad (4.1)$$

sujeito a:

$$\bar{\mathbf{x}}_T = \mathbf{x}_{goal} \quad (4.2)$$

$$\mathbf{x}_{t+1} = A\mathbf{x}_t + B\bar{\mathbf{u}}_t + \omega_t \quad \forall(t) \quad (4.3)$$

$$\mathbf{x}_0 \sim \mathcal{N}(\bar{\mathbf{x}}_0, \Sigma_{x_0}) \quad \omega_t \sim \mathcal{N}(0, \Sigma_{\omega_t}) \quad \forall(t) \quad (4.4)$$

$$Pr \left[ \left( \bigwedge_{j \in \mathbb{I}} \bigwedge_{t \in \mathcal{T}(\mathbb{I}_j)} \mathbf{x}_t \in \mathbb{I}_j \right) \wedge \left( \bigwedge_{j \in \mathbb{O}} \bigwedge_{t \in \mathcal{T}(\mathbb{O}_j)} \mathbf{x}_t \in \mathbb{O}_j \right) \right] \geq 1 - \Delta \quad (4.5)$$

A função objetivo (4.1) define que  $\mathcal{J}$  será minimizada no planejamento da missão, podendo ser relacionada com consumo de combustível, distância percorrida entre outros. Os autores em (BLACKMORE; ONO; WILLIAMS, 2011) definem a função objetivo como um custo proporcional a magnitude dos controles aplicados,  $\mathcal{J}(\cdot) = \sum_t \|\bar{\mathbf{u}}_t\|_2$ , sendo a função objetivo quadrática aproximada por um polígono de 32 faces.

A restrição (4.2) estabelece que no último instante o estado do veículo  $\bar{\mathbf{x}}_T$  deve coincidir com o estado  $\mathbf{x}_{goal}$  indicado para o final da missão. A transição dos estados do sistema é definida na restrição (4.3), em que o primeiro estado  $\mathbf{x}_0$  segue uma distribuição Gaussiana tendo um estado inicial esperado  $\bar{\mathbf{x}}_0$  e uma matriz de covariância  $\Sigma_{x_0}$ . Há também um ruído Gaussiano aditivo  $\omega_t$  com média zero e matriz de covariância  $\Sigma_{\omega_t}$ , aplicado a cada transição dos estados.

A restrição (4.5) estabelece que os estados  $\mathbf{x}_t$  devem respeitar os pontos de passagem ( $\mathbb{I}_j$ ) e estar fora dos obstáculos ( $\mathbb{O}_j$ ) em todos os instantes definidos na missão:  $t \in \mathcal{T}(\mathbb{I}_j)$  e  $t \in \mathcal{T}(\mathbb{O}_j)$ . Considerando o exemplo dado na Figura 7, teremos  $\mathcal{T}(\mathbb{I}_A) = \{3\}$ ,  $\mathcal{T}(\mathbb{I}_B) = \{5, 6, 7, 8\}$  e  $\mathcal{T}(\mathbb{O}_D) = \mathcal{T}(\mathbb{O}_E) = \{0, \dots, 12\}$ . Ainda na restrição (4.5), a expressão  $Pr[\cdot] \geq 1 - \Delta$  informa que a probabilidade de estar fora dos obstáculos, respeitando todos os pontos de passagem nos instantes definidos, deve ser maior ou igual a  $1 - \Delta$ . Logo, a probabilidade de falha é menor que  $\Delta$ .

Tanto os pontos de passagem quanto os obstáculos são definidos por regiões convexas. Os pontos de passagem são estabelecidos como junções de restrições lineares nas quais o VANT deve permanecer dentro, durante o intervalo de tempo  $t \in \mathcal{T}(\mathbb{I}_j)$  definido na missão. Por outro lado, os obstáculos são definidos como disjunções de restrições lineares nas quais o VANT deve permanecer fora de pelo menos uma delas durante  $t \in \mathcal{T}(\mathbb{O}_j)$ . As restrições (4.6) e (4.7) apresentam essa formalização, em que  $i \in H_j$  constituem os hiperplanos no espaço dos estados que definem as regiões convexas para obstáculos e pontos de passagem.

$$\mathbf{x}_t \in \mathbb{I}_j \Leftrightarrow \bigwedge_{i \in H_j^{\mathbb{I}}} h_i^T \mathbf{x}_t \leq g_i \quad (4.6)$$

$$\mathbf{x}_t \in \mathbb{O}_j \Leftrightarrow \bigvee_{i \in H_j^{\mathbb{O}}} h_i^T \mathbf{x}_t \geq g_i \quad (4.7)$$

Pelas restrições (4.6) e (4.7), percebe-se que os vetores normais  $h_i$  dos hiperplanos que definem as regiões convexas, sempre apontam para fora da região tanto para obstáculos quanto para pontos de passagem. Assim, intuitivamente, se desejamos estar dentro da região, teremos  $h_i^T \mathbf{x}_t \leq g_i$ ; se desejamos estar fora da região, teremos  $h_i^T \mathbf{x}_t \geq g_i$ . A Figura 8 exemplifica tais conceitos. Note que  $\mathbf{x}_t \in \mathbb{I}_j$  significa estar dentro do ponto de passagem e  $\mathbf{x}_t \in \mathbb{O}_j$  significa estar fora do obstáculo (ou seja, está respeitando a restrição).

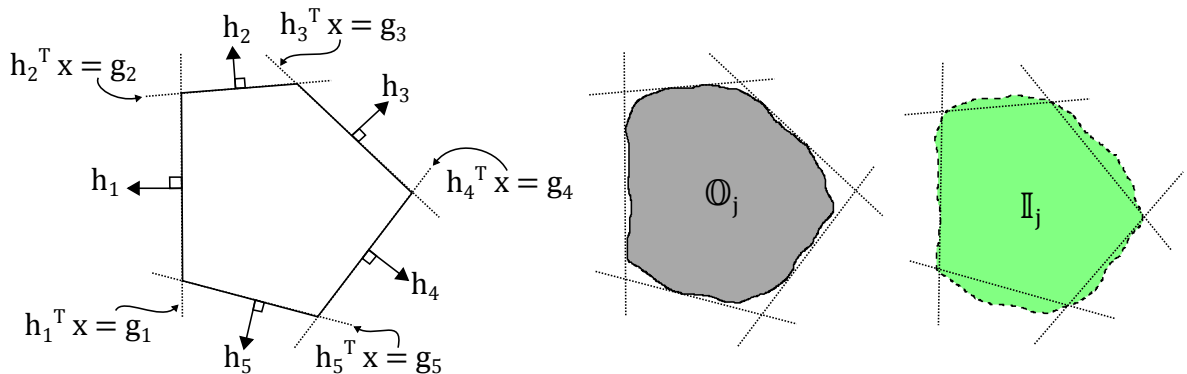


Figura 8 – Região convexa para modelar obstáculos  $\mathbb{O}_j$  e pontos de passagem  $\mathbb{I}_j$

### 4.3 Modelo determinístico para o problema

A formulação matemática apresentada para o CCPP revela que se trata de um problema de otimização estocástico, em que os estados  $\mathbf{x}_t$  são variáveis aleatórias. Uma formulação determinística foi proposta por (BLACKMORE; ONO; WILLIAMS, 2011) e será descrita a seguir.

Nesse problema é assumido que o estado inicial segue uma distribuição gaussiana  $\mathbf{x}_0 \sim \mathcal{N}(\bar{\mathbf{x}}_0, \Sigma_{x_0})$ , a transição dos estados do sistema segue uma equação de dinâmica linear e existe um ruído branco aditivo que modela os distúrbios, seguindo uma distribuição gaussiana  $\omega_t \sim \mathcal{N}(0, \Sigma_{\omega_t})$ . Logo, a distribuição dos estados futuros em qualquer instante  $t$  também será uma distribuição gaussiana  $\mathbf{x}_t \sim \mathcal{N}(\bar{\mathbf{x}}_t, \Sigma_t)$ . Os estados esperados  $\bar{\mathbf{x}}_t$  e covariância  $\Sigma_t$  futuros são definidos nas restrições 4.8 e 4.9, respectivamente, sendo obtidos ao resolver a recursão presente na restrição 4.3. A dedução dos estados esperados futuros apresentados na restrição 4.8 é direta, visto que por propriedade estatística as mesmas somas e multiplicações matriciais nas variáveis aleatórias são preservadas quando se considera seus valores esperados. Assim, visto que é conhecido o valor esperado inicial  $\bar{\mathbf{x}}_0$  é que a transição de estados adiciona um ruído gaussiano  $\omega_t$  cujo valor esperado é zero, o resultado será a equação 4.8. Por outro lado, a dedução da restrição 4.9 resulta da aplicação de duas propriedades estatísticas. A primeira diz que dado dois vetores  $x$  e  $z$  quaisquer de variáveis aleatórias com covariâncias  $\Sigma_x$  e  $\Sigma_z$ , a covariância de  $x + z$  é  $\Sigma_{xz} = \Sigma_x + \Sigma_z$ . A segunda propriedade diz que dado um vetor  $x$  qualquer de variáveis aleatórias e covariância  $\Sigma_x$ , a covariância de  $Ax$  é  $\Sigma_{Ax} = A \Sigma_x A^T$ .

$$\bar{\mathbf{x}}_{t+1} = A \bar{\mathbf{x}}_t + B \bar{\mathbf{u}}_t \quad \forall(t) \quad (4.8)$$

$$\Sigma_{t+1} = A \Sigma_t A^T + \Sigma_{\omega_t} \quad \forall(t) \quad (4.9)$$

Assim, assumindo que conhecemos o estado esperado no início  $\bar{\mathbf{x}}_0$  e dados os controles esperados  $\bar{\mathbf{u}}_t$ , é possível prever exatamente os estados esperados futuros  $\bar{\mathbf{x}}_t$ , que sabemos seguir uma distribuição gaussiana com uma covariância dada por  $\Sigma_t$ . Com isso em mente, é preciso garantir que a restrição 4.5 seja satisfeita dentro do risco especificado  $\Delta$ . Pelo teorema de De Morgan, estabelece-se que para um conjunto  $\{E_1, E_2, E_3, \dots\}$  de eventos quaisquer é válida a seguinte equivalência:

$$Pr \left[ \bigwedge_i E_i \right] = 1 - Pr \left[ \bigvee_i \neg E_i \right] \quad (4.10)$$

Desta forma, a restrição 4.5 pode ser reescrita conforme a restrição 4.11. Intuitivamente, isto equivale a dizer que a probabilidade de violar regiões de passagem  $\mathbf{x}_t \notin \mathbb{I}_j$  ou

colidir com obstáculos  $\mathbf{x}_t \notin \mathbb{O}_j$  deve ser menor do que  $\Delta$ .

$$Pr \left[ \left( \bigvee_{j \in \mathbb{I}} \bigvee_{t \in \mathcal{T}(\mathbb{I}_j)} \mathbf{x}_t \notin \mathbb{I}_j \right) \vee \left( \bigvee_{j \in \mathbb{O}} \bigvee_{t \in \mathcal{T}(\mathbb{O}_j)} \mathbf{x}_t \notin \mathbb{O}_j \right) \right] \leq \Delta \quad (4.11)$$

A restrição será decomposta utilizando a desigualdade de Boole, em que para um conjunto qualquer de eventos  $E_i$  é válida a seguinte desigualdade:

$$Pr \left[ \bigvee_i E_i \right] \leq \sum_i Pr [E_i] \quad (4.12)$$

Assim, pode-se garantir o atendimento da restrição 4.11 através da restrição 4.13, ou seja, a restrição 4.13 implica na restrição 4.11 e a nova restrição implicará em um modelo determinístico mais conservador em relação ao modelo estocástico original.

$$\sum_{j \in \mathbb{I}} \sum_{t \in \mathcal{T}(\mathbb{I}_j)} Pr [\mathbf{x}_t \notin \mathbb{I}_j] + \sum_{j \in \mathbb{O}} \sum_{t \in \mathcal{T}(\mathbb{O}_j)} Pr [\mathbf{x}_t \notin \mathbb{O}_j] \leq \Delta \quad (4.13)$$

Seguindo o mesmo princípio e tomando a definição das regiões  $\mathbb{I}$  e  $\mathbb{O}$  dadas nas equações 4.6 e 4.7, as seguintes equivalências são obtidas:

$$Pr [\mathbf{x}_t \notin \mathbb{I}_j] \Leftrightarrow Pr \left[ \bigvee_{i \in H_j^{\mathbb{I}}} h_i^T \mathbf{x}_t > g_i \right] \leq \sum_{i \in H_j^{\mathbb{I}}} Pr [h_i^T \mathbf{x}_t > g_i] \quad (4.14)$$

$$Pr [\mathbf{x}_t \notin \mathbb{O}_j] \Leftrightarrow Pr \left[ \bigwedge_{i \in H_j^{\mathbb{O}}} h_i^T \mathbf{x}_t < g_i \right] \quad (4.15)$$

Isso permite expressar as restrições de probabilidades individuais em termos dos valores esperados dos estados  $\bar{\mathbf{x}}_t$  e covariância  $\Sigma_t$  como descrito pelas restrições 4.16 e 4.17. Uma função  $m_{t,i}(\cdot)$  é utilizada para fazer a alocação dos riscos ( $\delta_{ti}^{\mathbb{I}}$  e  $\delta_{tj}^{\mathbb{O}}$ ), permitindo reescrever a restrição 4.13 em termos das alocações como estabelecido na restrição 4.18.

$$Pr [h_i^T \mathbf{x}_t > g_i] \leq \delta_{ti}^{\mathbb{I}} \Leftrightarrow h_i^T \bar{\mathbf{x}}_t \leq g_i - m_{t,i}(\delta_{ti}^{\mathbb{I}}) \quad (4.16)$$

$$Pr \left[ \bigwedge_{i \in H_j^{\mathbb{O}}} h_i^T \mathbf{x}_t < g_i \right] \leq \delta_{tj}^{\mathbb{O}} \Leftrightarrow \bigvee_{i \in H_j^{\mathbb{O}}} [h_i^T \bar{\mathbf{x}}_t \geq g_i + m_{t,i}(\delta_{tj}^{\mathbb{O}})] \quad (4.17)$$

$$\sum_{j \in \mathbb{I}} \sum_{t \in \mathcal{T}(\mathbb{I}_j)} \sum_{i \in H_j^{\mathbb{I}}} \delta_{ti}^{\mathbb{I}} + \sum_{j \in \mathbb{O}} \sum_{t \in \mathcal{T}(\mathbb{O}_j)} \delta_{tj}^{\mathbb{O}} \leq \Delta \quad (4.18)$$

A disjunção presente na restrição 4.17 é resolvida definindo uma nova variável binária  $Y_{ti}$  e aplicando as regras lógicas (S2) e (R9) das tabelas 6 e 5 que resultam nas restrições 4.19 e 4.20.

$$Y_{ti} \Rightarrow h_i^T \bar{\mathbf{x}}_t \geq g_i + m_{t,i}(\delta_{jt}^{\circ}) \quad \forall (j, t \in \mathcal{T}(\mathbb{O}_j), i \in H_j^{\circ}) \quad (4.19)$$

$$\sum_{i \in H_j^{\circ}} Y_{ti} \geq 1 \quad \forall (j, t \in \mathcal{T}(\mathbb{O}_j)) \quad (4.20)$$

O risco de violar cada restrição é alocado pela função  $m_{t,i}(\cdot)$  que é uma função contínua, tal como definida na equação 4.21. Essa função aloca o risco  $\xi$  de se estar do lado oposto ao desejado em um hiperplano  $i$  qualquer. Intuitivamente, isso é o mesmo que encontrar o ponto na função inversa da distribuição acumulada ( $cdf^{-1}$ ), no qual teremos  $(1 - \xi)$  de certeza que  $\mathbf{x}_t$  estará no lado desejado. A Figura 9 ilustra graficamente todo esse processo para regiões do tipo  $\mathbb{I}$  e  $\mathbb{O}$ . O termo  $\sqrt{h_i^T \Sigma_t h_i}$  na equação 4.21 é o desvio padrão  $\sigma$  da distribuição gaussiana quando é feita a projeção da covariância  $\Sigma_t$  na direção do vetor normal ao hiperplano  $h_i$ . Para que a equação 4.21 faça corretamente a alocação do erro, é preciso que o vetor normal  $h_i$  esteja normalizado ( $\|h_i\| = 1$ ) antes do modelo ser codificado. Caso isso não seja feito, haverá problemas de escala onde as rotas projetadas podem ser conservativas quando  $\|h_i\| > 1$  ou, mesmo, violar o risco quando  $\|h_i\| < 1$ .

$$m_{t,i}(\xi) = cdf^{-1}(1 - \xi) \sqrt{h_i^T \Sigma_t h_i} \quad (4.21)$$

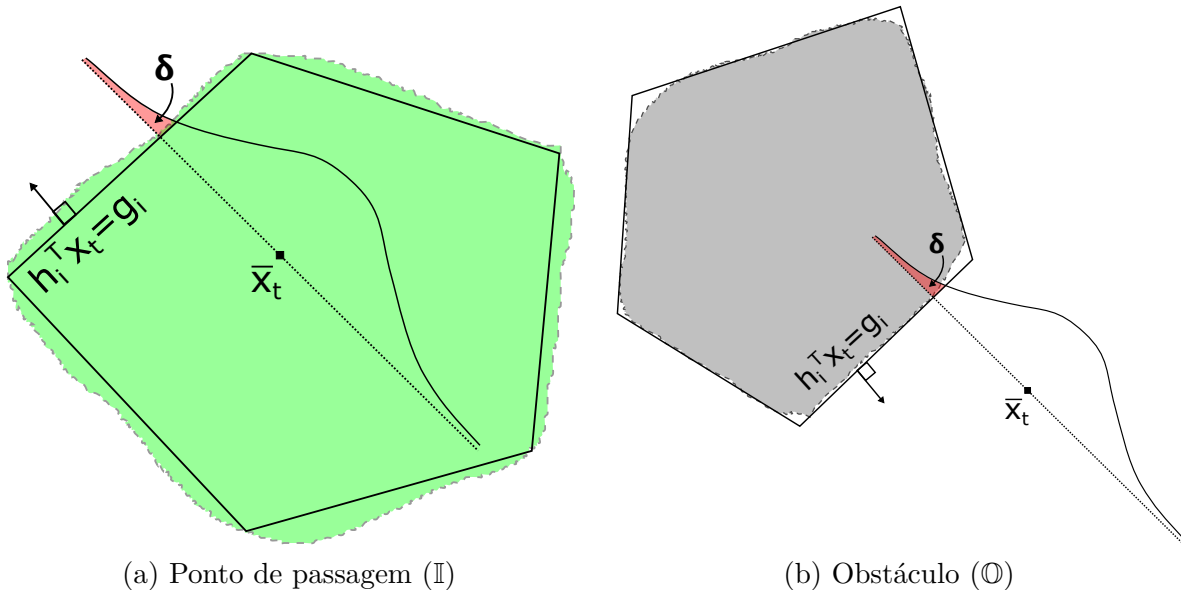


Figura 9 – Calculando a probabilidade de violar regiões do tipo  $\mathbb{I}$  e  $\mathbb{O}$

Em termos práticos não há uma forma analítica de se calcular  $cdf^{-1}$ , sendo seus valores obtidos por tabela ou algoritmos iterativos. Normalmente, obtém-se a  $cdf^{-1}$  através de algoritmos que calculam a função inversa do erro  $erf^{-1}$ , no qual a função do erro  $erf(z)$  é definida pela equação (4.22). A relação entre  $cdf^{-1}$  e  $erf^{-1}$  é estabelecida na equação 4.23.

$$erf(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt \quad (4.22)$$

$$cdf^{-1}(1 - \xi) = erf^{-1}(1 - 2\xi) \cdot \sqrt{2} \quad (4.23)$$

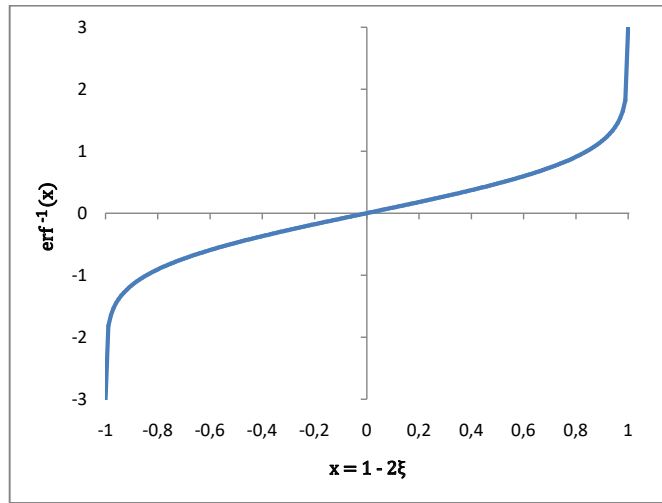


Figura 10 – Gráfico da função inversa do erro:  $erf^{-1}$

A Figura 10 apresenta a curva para a função inversa do erro. Nota-se que  $erf^{-1}$  é não linear e não convexa, mas restringindo o domínio  $0 < \xi \leq \Delta \leq 0.5$  a função  $erf^{-1}$  se torna convexa e pode ser resolvida por *solvers* não lineares de forma ótima. Assim, o problema completo também pode ser resolvido de forma ótima ao combinar o método branch-and-bound para explorar a disjunção presente na equação 4.17 e resolver a alocação do risco nos subproblemas relacionados usando *solvers* não lineares. A restrição adicional estabelecida para  $\Delta \leq 0.5$  não inviabiliza aplicações práticas, pois em geral é desejado que a probabilidade de falha seja bem menor do que 50%. A formulação completa determinística para o CCPP é apresentada a seguir:

$$\text{Minimizar } \mathcal{J}(\cdot) \quad (4.1)$$

$$\bar{\mathbf{x}}_T = \mathbf{x}_{goal} \quad (4.2)$$

$$\bar{\mathbf{x}}_{t+1} = A \bar{\mathbf{x}}_t + B \bar{\mathbf{u}}_t \quad \forall(t) \quad (4.8)$$

$$h_i^T \bar{\mathbf{x}}_t \leq g_i - m_{t,i}(\delta_{ti}^{\mathbb{I}}) \quad \forall (j, t \in \mathcal{T}(\mathbb{I}_j), i \in H_j^{\mathbb{I}}) \quad (4.24)$$

$$h_i^T \bar{\mathbf{x}}_t \geq g_i + m_{t,i}(\delta_{jt}^{\mathbb{O}}) + M(Y_{ti} - 1) \quad \forall (j, t \in \mathcal{T}(\mathbb{O}_j), i \in H_j^{\mathbb{O}}) \quad (4.25)$$

$$\sum_{i \in H_j^{\mathbb{O}}} Y_{ti} \geq 1 \quad \forall (j, t \in \mathcal{T}(\mathbb{O}_j)) \quad (4.20)$$

$$\sum_{j \in \mathbb{I}} \sum_{t \in \mathcal{T}(\mathbb{I}_j)} \sum_{i \in H_j^{\mathbb{I}}} \delta_{ti}^{\mathbb{I}} + \sum_{j \in \mathbb{O}} \sum_{t \in \mathcal{T}(\mathbb{O}_j)} \delta_{tj}^{\mathbb{O}} \leq \Delta \quad (4.18)$$

$$\bar{\mathbf{x}}_t \in \mathbb{X} \subseteq \mathbb{R}^n, \quad \bar{\mathbf{u}}_t \in \mathbb{U} \subseteq \mathbb{R}^m \quad (4.26)$$

$$(\delta_{ti}^{\mathbb{I}}, \delta_{tj}^{\mathbb{O}}) \geq 0 \quad (4.27)$$

$$Y_{ti} \in \{0, 1\} \quad (4.28)$$

onde:

$$\Sigma_{t+1} = A \Sigma_t A^T + \Sigma_{w_t} \quad (4.9)$$

$$m_{t,i}(\xi) = \text{erf}^{-1}(1 - 2\xi) \sqrt{2h_i^T \Sigma_t h_i} \quad (4.29)$$

A restrição 4.24 é derivada diretamente da restrição 4.16 e a restrição 4.25 é derivada da restrição 4.19 ao se aplicar a regra lógica (S2) da tabela 6.

## 4.4 Discussão a respeito das hipóteses do modelo

Uma questão importante para a aplicabilidade do modelo proposto é a suposição de que as incertezas podem ser modeladas como uma distribuição gaussiana. Para verificar se essa hipótese é consistente com nossa aplicação foram coletados dados de 32 horas de voo em simulador, no qual foi medido o quanto a aeronave se desviava da rota planejada. A Figura 11 apresenta a distribuição do erro na posição da aeronave em relação a rota, onde a curva ilustra qual seria a distribuição gaussiana (ou distribuição normal) para o mesmo conjunto de dados. Com base na figura, pode-se afirmar que a distribuição não é uma gaussiana perfeita. Esse fato também é confirmado ao se aplicar testes para verificar a normalidade dos dados. Entretanto, a distribuição dos dados é suficientemente próxima



de uma normal para manter válido o nível de risco  $\Delta$  por dois motivos. O primeiro é que a distribuição se mostrou mais centralizada que a normal, o que auxilia em um risco de fato menor do que o especificado, caso sejam escolhidos valores baixos como  $\Delta \leq 0.16$  (1 desvio padrão) ou inferiores. O segundo motivo está no nível de conservantismo imposto pela desigualdade de Boole utilizada na construção do modelo determinístico (equação 4.12). Assim, na prática tal conservantismo acaba absorvendo as imperfeições na suposição de uma distribuição normal.

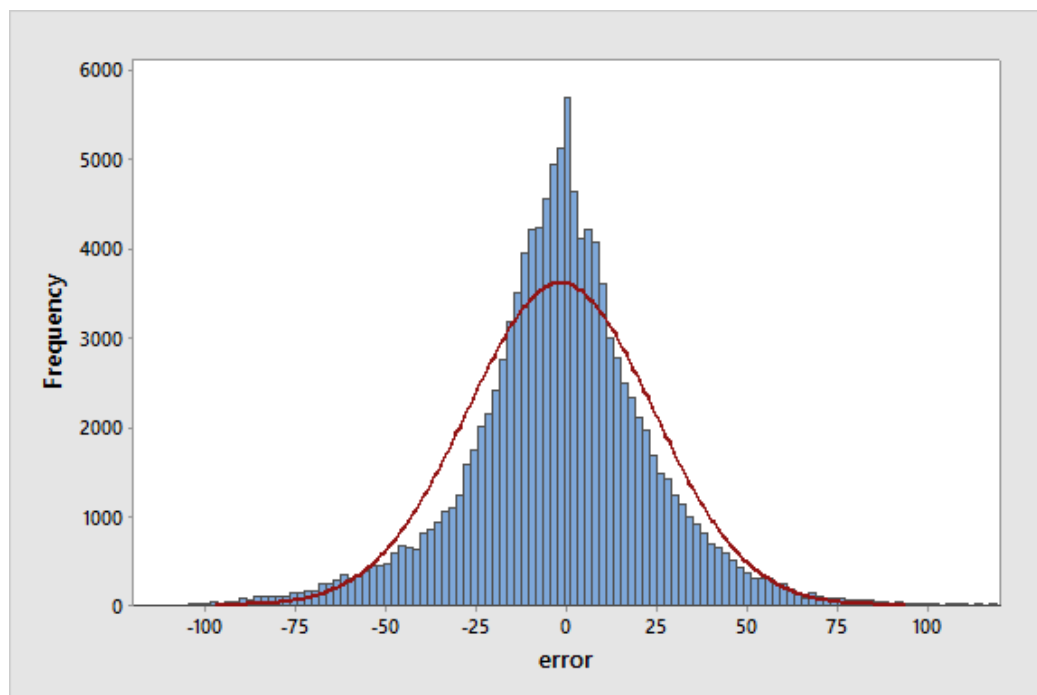


Figura 11 – A posição do VANT relativo a rota aproxima-se de uma distribuição gaussiana

O trabalho também assume que os obstáculos e os pontos de passagem são regiões convexas, apesar disso não há nenhuma limitação para aplicações onde tais regiões sejam não convexas, uma vez que é possível modelar tais regiões como um conjunto de regiões convexas, como ilustrado na Figura 12. Nota-se que as 6 regiões convexas da imagem estão conectadas formando, assim, uma única região não convexa.

Por fim, o trabalho define uma equação de transição dos estados, como uma equação linear em termos dos estados e controles. Na realidade, os veículos nos quais tais modelos são aplicados não seguem uma dinâmica linear e a literatura ainda carece de técnicas que consigam resolver na otimalidade um problema como esse, que considere a dinâmica real desses veículos. Entretanto, tal equação linear é amplamente utilizada no controle de diversos sistemas e tem retornado resultados válidos na prática, mesmo quando a hipótese de linearidade do sistema é violada. Como exemplo, tem-se visto o uso do filtro de Kalman, projetado para sistemas lineares mas, ainda assim, é utilizado com sucesso para sistemas robóticos não lineares. Os resultados obtidos nesta tese também comprovaram o mesmo a partir dos testes com simuladores de voo realizados e descritos no capítulo 7.

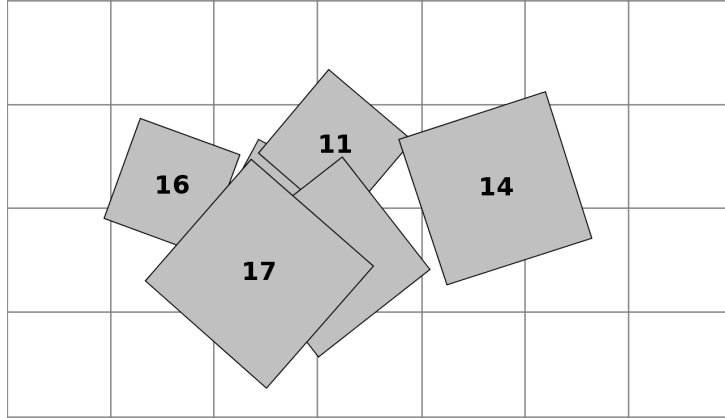


Figura 12 – Obstáculo não convexo formado por 6 regiões convexas

## 4.5 Modelo linear para o problema

Esta seção apresenta a primeira contribuição desta tese de doutorado ao propor um modelo matemático linear inteiro-misto a partir do modelo não-linear inteiro-misto introduzido em (BLACKMORE; ONO; WILLIAMS, 2011). A presente seção irá introduzir algumas funções objetivos lineares e não-lineares, bem como descreverá linearizações para funções não lineares, como as funções norma-2, quadrática e inversa do erro.

No modelo determinístico apresentado até o momento, não foi apresentado explicitamente qual função objetivo a ser otimizada. Essa função foi descrita, genericamente, pela expressão 4.1. Pode-se definir funções que minimizem vários tipos de critérios: distância percorrida, consumo de combustível, tempo, entre outros. As funções objetivos  $\mathcal{J}(\cdot)$  mais gerais e comuns na literatura se aplicam ao vetor de controles como estabelecido nas equações 4.30 - 4.32.

$$\mathcal{J}(\mathbf{u}_t) = \sum_t \|\mathbf{u}_t\|_1 \quad (4.30)$$

$$\mathcal{J}(\mathbf{u}_t) = \sum_t \|\mathbf{u}_t\|_2 \quad (4.31)$$

$$\mathcal{J}(\mathbf{u}_t) = \sum_t \mathbf{u}_t^T \cdot \mathbf{u}_t \quad (4.32)$$

A equação 4.30 minimiza a soma da norma-1 dos controles, sendo uma equação que pode ser definida diretamente em termos lineares ao considerar  $\mathbf{u}_t^d$  a  $d$ -ésima componente

do vetor de controles  $\mathbf{u}_t$ . Isso é apresentado formalmente na equação 4.33.

$$\left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \|\mathbf{u}_t\|_1 \right\} \Leftrightarrow \left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \sum_d |\mathbf{u}_t^d| \right\} \Leftrightarrow \left\{ \begin{array}{l} \mathcal{J}(\mathbf{u}_t) = \sum_t \sum_d r_t^d \\ r_t^d \geq +\mathbf{u}_t^d \\ r_t^d \geq -\mathbf{u}_t^d \end{array} \right\} \quad (4.33)$$

A equação 4.31 minimiza a norma-2 dos controles cuja definição é não linear conforme apresentado na equação 4.34.

$$\left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \|\mathbf{u}_t\|_2 \right\} \Leftrightarrow \left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \sqrt{\sum_d |\mathbf{u}_t^d|^2} \right\} \quad (4.34)$$

Entretanto, os autores em (BLACKMORE; ONO; WILLIAMS, 2011) apresentam uma formulação aproximada, baseada em um polígono de 32 faces, considerando casos específicos no plano 2D, em que o vetor de controle possui apenas duas componentes  $d \in \{x, y\}$ . Essa ideia é apresentada na Figura 13 e formalizada na equação 4.35 com  $n \in \{0, \dots, N-1\}$

$$\left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \sqrt{\sum_d |\mathbf{u}_t^d|^2} \right\} \approx \left\{ \begin{array}{l} \mathcal{J}(\mathbf{u}_t) = \sum_t r_t \\ r_t \geq \cos\left(\frac{2\pi n}{N}\right) u_t^x + \sin\left(\frac{2\pi n}{N}\right) u_t^y \quad \forall(t, n) \end{array} \right\} \quad (4.35)$$

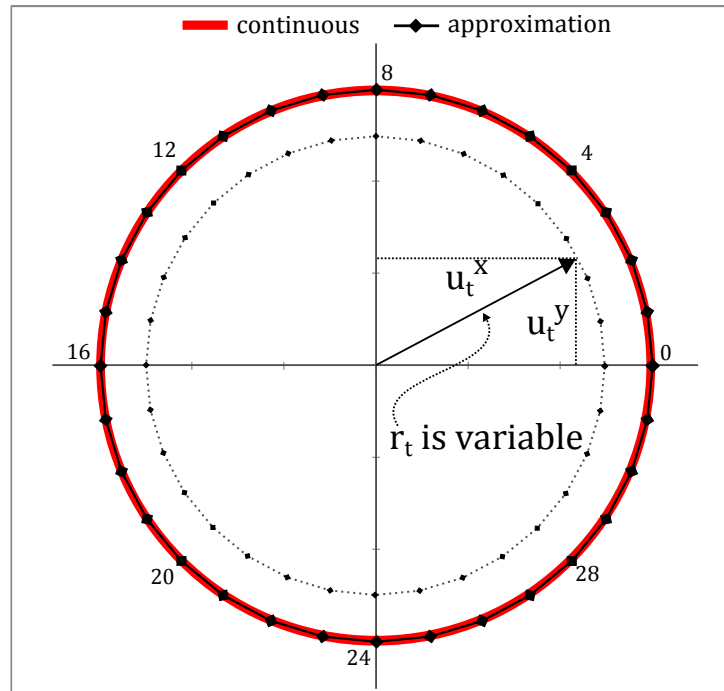


Figura 13 – Aproximação da norma-2 por um polígono de 32 faces.

A equação 4.32 minimiza o produto escalar dos controles e permite codificar o problema diretamente como um problema de programação quadrática inteira mista (PQIM).

A resolução de um PQIM costuma ser mais custosa em geral do que um PLIM, assim é proposta uma aproximação linear para a função quadrática como uma função linear por partes. O domínio da função passa a ser subdividido em  $N$  partes lineares como ilustrado na Figura 14a e formalizado na equação 4.36. Onde  $\lambda_{tn}^d$  são variáveis auxiliares e  $M_n$  é o coeficiente de crescimento do  $n$ -ésimo seguimento de reta que modela a curva.

$$\left\{ \mathcal{J}(\mathbf{u}_t) = \sum_t \mathbf{u}_t^T \cdot \mathbf{u}_t \right\} \approx \left\{ \begin{array}{l} \mathcal{J}(\mathbf{u}_t) = \sum_t \sum_d \sum_n M_n \lambda_{tn}^d \\ +u_t^d \leq \sum_n \lambda_{tn}^d \\ -u_t^d \leq \sum_n \lambda_{tn}^d \\ 0 \leq \lambda_{tn}^d \leq \frac{u^{max}}{N} \\ \text{-----} \\ M_n = \frac{f(n+1)-f(n)}{x(n+1)-x(n)} = (2n+1) \frac{u^{max}}{N} \end{array} \right. \quad (4.36)$$

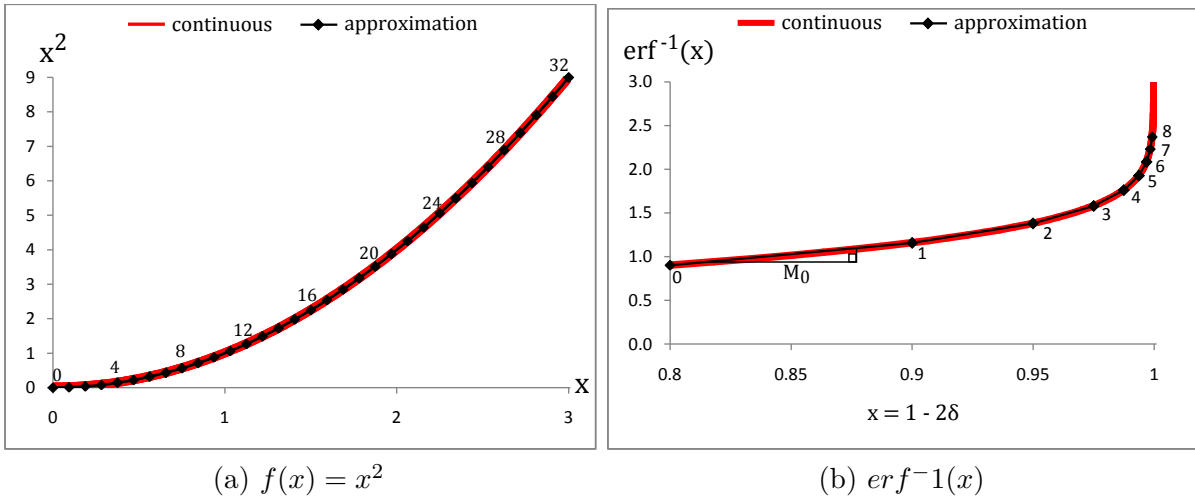


Figura 14 – Aproximações por funções lineares por partes

Por último, a mesma ideia de linearização é proposta para aproximar a função inversa do erro, como uma função linear por partes. Esse processo é ilustrado na Figura 14b e formalizado na equação 4.36.

$$\left\{ m_{t,i}(\xi) = erf^{-1}(1 - 2\xi) \sqrt{2h_i^T \Sigma_t h_i} \right\} \approx \left\{ \begin{array}{l} m_{t,i}(\xi) = (f(0) + \sum_n M_n \lambda_n) \sqrt{2h_i^T \Sigma_t h_i} \\ 1 - 2\xi = x(0) + \sum_n \lambda^n \\ 0 \leq \lambda_n \leq x(n+1) - x(n) \\ \text{-----} \\ M_n = \frac{f(n+1)-f(n)}{x(n+1)-x(n)} \end{array} \right. \quad (4.37)$$

# CHANCE-CONSTRAINED QUALITATIVE STATE PLAN PROBLEM

## 5.1 Introdução

O *Chance-Constrained Qualitative State Plan problem* (CCQSP) é introduzido em (ONO; WILLIAMS; BLACKMORE, 2013), sendo representado como um DAG (*directed acyclic graph*). A Figura 15 apresenta um exemplo do CCQSP e uma possível solução considerando a missão descrita a seguir.

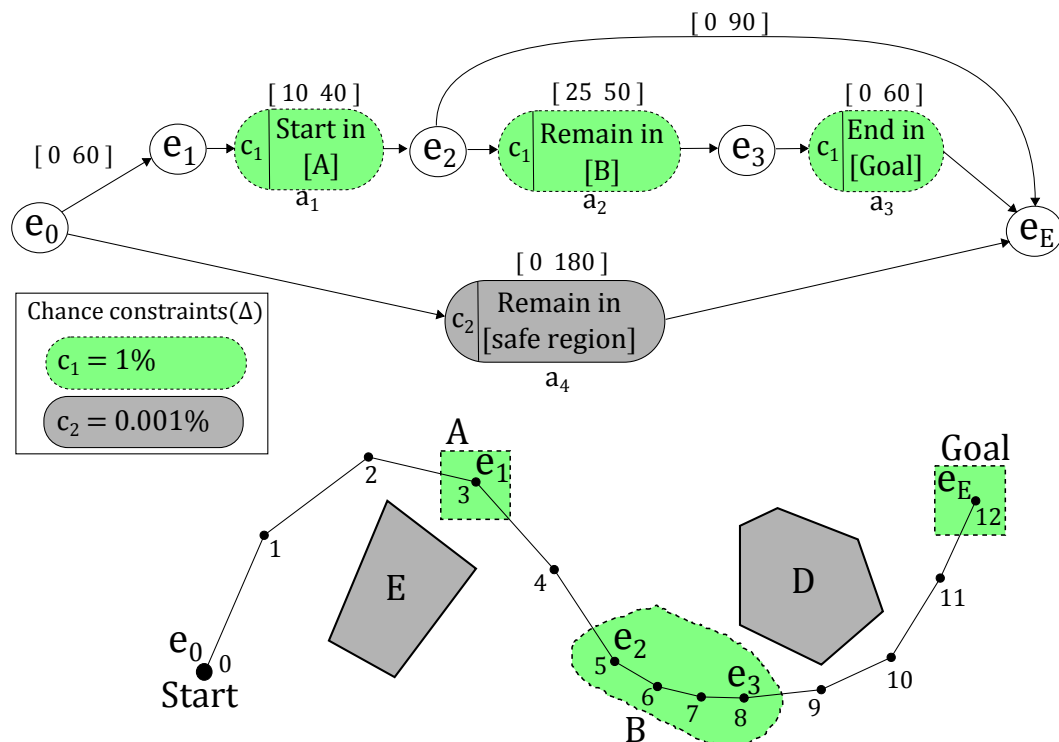


Figura 15 – Um exemplo de um CCQSP e sua solução considerando uma discretização do horizonte de planejamento de 10 segundos

Iniciando no ponto **Start** o VANT deve alcançar a região **A**, em no máximo 1 minuto. Em seguida, partindo de **A** deve chegar em **B**, entre 10 e 40 segundos, permanecendo lá entre 25 a 50 segundos. A missão termina na região **Goal**, com mais 1 minuto de prazo. A probabilidade de falha no cumprimento dessas etapas da missão é limitada em 1%. O tempo total para cumprir a missão é de 3 minutos e o tempo decorrido entre os eventos  $e_2$  e  $e_E$  deve ser de no máximo 1.5 minuto. O VANT deve permanecer na região segura durante todo o tempo da missão, evitando colisão com os obstáculos **D** e **E** dentro de uma margem de risco de 0.001%.

Os vértices do DAG na figura 15 é composto por eventos (vértices) e marcam os instantes de tempo entre cada etapa da missão (arcos). As etapas da missão são chamadas de episódios e definem a tarefa do VANT entre dois eventos quaisquer. No trabalho descrito em (ONO; WILLIAMS; BLACKMORE, 2013), os autores concentraram seus esforços em fazer a união do tratamento da incerteza (*chance-constrained*) com o *scheduling* (sequenciamento), considerando os tempos dos eventos, definindo, assim, o problema CCQSP. Os mesmos autores apresentam um algoritmo planejador chamado *p-Sulu Planner* que utiliza *solvers* comerciais para solucionar as *chance-constrained*, onde uma especialização do algoritmo *branch-and-bound*, proposta pelos autores, fica responsável por encontrar o *scheduling* ótimo.

A contribuição de nosso trabalho está em resolver o CCQSP completamente com *solvers* comerciais, utilizando o algoritmo *branch-and-cut*. O algoritmo *branch-and-cut* é um método eficiente que une as vantagens do algoritmo *branch-and-bound* com o método de plano de cortes. Como consequência dessa inovação, conseguiu-se reduzir o tempo computacional para resolver o problema conforme será reportado pelos resultados obtidos. Esse objetivo foi alcançado com a proposta de um modelo completo e fortalecido para o CCQSP, que pela primeira vez foi codificado em um único modelo de programação linear inteira-misto, contendo as *chance-constrained* e o *scheduling* do problema simultaneamente.

## 5.2 Descrição do problema

O *CCQSP Planning Problem* é formalmente definido em (ONO; WILLIAMS; BLACKMORE, 2013), onde o horizonte de planejamento é discretizado em um conjunto de instantes  $t = 0, 1, \dots, T$  com duração  $\Delta t$  entre dois instantes consecutivos.

**Definição 1** (CCQSP Planning Problem). Dado  $\langle \mathbf{x}, \mathbf{u}, \mathcal{I}, \mathcal{M}, \mathcal{P}, \mathcal{J} \rangle$ , o *CCQSP Planning Problem* deve encontrar uma solução ótima  $\mathcal{S}^{**}$  onde:

- $\mathbf{x}$  é o conjunto de variáveis que representam o estado do veículo ( $\mathbf{x}_t$ ) a cada instante de tempo  $t$ .

- $\mathbf{u}$  é o conjunto de variáveis que representam o controle ( $\mathbf{u}_t$ ) aplicado ao veículo a cada instante de tempo  $t$ .
- $\mathcal{I} = \langle \hat{\mathbf{x}}_0, \Sigma_{x_0} \rangle$  é as condições iniciais do veículo (Def. 2).
- $\mathcal{M} = \langle A, B, \Sigma_{w_t} \rangle$  é o modelo estocástico do veículo (Def. 3).
- $\mathcal{P}$  é o *Chance-Constrained Qualitative State Plan* da missão, especificando as etapas a serem cumpridas, sua ordem e os limites de tempo dados a cada etapa, bem como as restrições de risco envolvidas. (Def. 4).
- $\mathcal{J}$  é uma função objetivo a ser otimizada na missão.
- $\mathcal{S}^{**} = \langle \mathbf{x}^*, \mathbf{u}^*, s^* \rangle$  é uma sequência ótima das variáveis de estado  $\mathbf{x}^*$ , dos controles  $\mathbf{u}^*$  e o *scheduling* ótimo  $s^*$ .

Por exemplo, considerando um cenário representado no espaço 2D como ilustrado na Figura 15, vamos supor que  $\mathbf{x} := [p^x, p^y, v^x, v^y]^T$  e  $\mathbf{u} := [a^x, a^y]^T$ . Neste contexto, temos os estados dados em função das posições e velocidades no eixo  $(x, y)$  como  $[p^x, p^y]$  e  $[v^x, v^y]$ , respectivamente. Os controles são determinados pelas acelerações aplicadas  $[a^x, a^y]$  nos eixos  $(x, y)$ .

**Definição 2** (Condições Iniciais).  $\mathcal{I} = \langle \hat{\mathbf{x}}_0, \Sigma_{x_0} \rangle$ , onde:

- $\bar{\mathbf{x}}_0$  é o estado esperado do veículo no início do planejamento ( $t = 0$ ).
- $\Sigma_{x_0}$  é a matriz de covariância para o estado do veículo em  $\mathbf{x}_0$ , assumindo uma distribuição normal, tal que:  $\mathbf{x}_0 \sim \mathcal{N}(\bar{\mathbf{x}}_0, \Sigma_{x_0})$

**Definição 3** (Modelo Estocástico do Veículo).  $\mathcal{M} = \langle A, B, \Sigma_{w_t} \rangle$  descrito pelas equações 5.1 e 5.2:

$$\mathbf{x}_{t+1} = A\mathbf{x}_t + B\mathbf{u}_t + \omega_t \quad \forall(t) \quad (5.1)$$

$$\mathbf{x}_t \in \mathbb{X}, \quad \mathbf{u}_t \in \mathbb{U}, \quad \omega_t \sim \mathcal{N}(0, \Sigma_{w_t}) \quad \forall(t) \quad (5.2)$$

Os estados  $\mathbf{x}_t \sim \mathcal{N}(\bar{\mathbf{x}}_t, \Sigma_t)$  e os controles  $\mathbf{u}_t \sim \mathcal{N}(\bar{\mathbf{u}}_t, K_t \Sigma_t)$  são variáveis aleatórias,  $\bar{\mathbf{x}}_t$  e  $\bar{\mathbf{u}}_t$  são os valores esperados para os controles e estados, respectivamente, a cada instante de tempo  $t$  com covariância  $\Sigma_t$ . Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) implementam um controle em malha fechada. Neste tipo de controle foi suposto que a aeronave é dotada de um sensor, como o GPS, capaz de mensurar a posição e, assim, é possível que correções online sejam feitas nos controles  $\mathbf{u}_t$ . O resultado disso é o amortecimento no crescimento da incerteza  $\Sigma_t$  no estado do veículo ao longo do tempo. A equação 5.3 estabelece essa correção, onde  $K_t$  é uma matriz de *feedback*, que precisa

ser escolhida apropriadamente para amortecer o crescimento da incerteza  $\Sigma_t$  dada pela equação 5.4:

$$\mathbf{u}_t = \bar{\mathbf{u}}_t + K_t(\mathbf{x}_t - \bar{\mathbf{x}}_t) \quad \forall(t) \quad (5.3)$$

$$\Sigma_{t+1} = (A + BK_t) \Sigma_t (A + BK_t)^T + \Sigma_{\omega_t} \quad \forall(t) \quad (5.4)$$

Na equação 5.2,  $\mathbf{x}_t \in \mathbb{X}$  e  $\mathbf{u}_t \in \mathbb{U}$  são, respectivamente, dois sub-espacos convexos definindo o domínio dos estados e controles dentro dos limites válidos para um dado VANT. Por exemplo,  $\mathbb{X}$  e  $\mathbb{U}$  poderiam definir limites de velocidade e aceleração possíveis para um VANT. A variável aleatória e independente do estado  $\omega_t$  representa distúrbios causados pela incerteza do ambiente. Assume-se que o distúrbio possa ser modelado como um ruído Gaussiano com média zero e matriz de covariância  $\Sigma_{x_0}$ . Na equação 5.1, as matrizes  $A$  e  $B$  definem a transição dos estados. As equações 5.5 e 5.6 exemplificam a utilização das equações 5.1 e 5.2, respectivamente, considerando o cenário 2D ilustrado na Figura 15.

$$\begin{bmatrix} p^x \\ p^y \\ v^x \\ v^y \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p^x \\ p^y \\ v^x \\ v^y \end{bmatrix}_t + \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a^x \\ a^y \end{bmatrix}_t + \omega_t \quad (5.5)$$

$$\begin{aligned} \mathbf{x}_t \in \mathbb{X} &\Leftrightarrow \begin{bmatrix} (p^x, p^y) \in \mathbb{R}^2 \\ \|v^x, v^y\| \leq v^{max} \end{bmatrix} \\ \mathbf{u}_t \in \mathbb{U} &\Leftrightarrow \begin{bmatrix} \|a^x, a^y\| \leq a^{max} \end{bmatrix} \end{aligned} \quad (5.6)$$

**Definição 4** (Chance-Constrained Qualitative State Plan).  $\mathcal{P} = \langle \mathcal{E}, \mathcal{T}, \mathcal{A}, \mathcal{C} \rangle$ , onde:

- $\mathcal{E} = \{e_0, e_1, \dots\}$  é um conjunto de eventos discretos, cada evento  $e \in \mathcal{E}$  representa um instante no tempo.
- $\mathcal{T} = \{r_1, r_2, \dots\}$  são as restrições temporais entre os eventos, indicando os tempos mínimos e máximos de duração entre dois eventos quaisquer:  $[\mathcal{T}_r^{lb}, \mathcal{T}_r^{ub}]$ .
- $\mathcal{A} = \{a_1, a_2, \dots\}$  é um conjunto de episódios, onde cada episódio  $a \in \mathcal{A}$  é definido entre dois eventos quaisquer. Os episódios definem a ordem desejada no cumprimento da missão, indicando as restrições que devem ser respeitadas para cada episódio da missão (*Def. 5*).
- $\mathcal{C} = \{c_1, c_2, \dots\}$  é o conjunto de *chance constraints*  $c \in \mathcal{C}$ .

No exemplo da Figura 15, temos:



1.  $\mathcal{E} = \{e_0, e_1, e_2, e_3, e_E\}$  (cinco eventos).
2.  $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$  (quatro episódios).
3.  $\mathcal{T} = \{[0\ 60], [10\ 40], [25\ 50], [0\ 60], [0\ 90], [0\ 180]\}$  (seis restrições temporais).
4.  $\mathcal{C} = \{c_1, c_2\}$  (duas *chance constraints*).

Uma restrição temporal  $r \in \mathcal{T}$  é estabelecida entre um evento de início ( $e_r^S$ ) e de fim ( $e_r^E$ ), informando um limite inferior ( $\mathcal{T}_r^{lb}$ ) e superior ( $\mathcal{T}_r^{ub}$ ) de tempo que o VANT deve respeitar para o cumprimento da missão. Cada episódio também é associado a uma *chance constraint* ( $c \in \mathcal{C}$ ), onde o conjunto  $\Psi_c$  denota todos os episódios associados a cada *chance constraint* e  $\Delta_c$  denota o risco máximo que pode ser atribuído a todos os episódios  $a \in \Psi_c$ . Assim, no exemplo da Figura 15, temos:  $\Psi_{c_1} = \{a_1, a_2, a_3\}$ ,  $\Psi_{c_2} = \{a_4\}$  e  $\Delta_{c_1} = 1\%$ ,  $\Delta_{c_2} = 0.001\%$ .

**Definição 5** (Episódios). Cada episódio  $a = \langle e_a^S, e_a^E, \Pi_a, R_a \rangle$  é composto por:

- $e_a^S$  um evento inicial do episódio.
- $e_a^E$  um evento final do episódio.
- $\Pi_a$  um conjunto de todos os instantes  $t$  no qual o episódio está ativo.
- $R_a$  um conjunto de restrições ( $x_t \in R_a$ ) do episódio que deve ser obedecido durante os instantes  $t \in \Pi_a$ .

Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) estabelecem três tipos de episódios de interesse  $\mathcal{A}$ :

1. *Start-in* ( $a \in \mathcal{A}^S$ ): implica  $x_t \in R_a$  no início do episódio.
2. *End-in* ( $a \in \mathcal{A}^E$ ): implica  $x_t \in R_a$  no fim do episódio.
3. *Remain-in* ( $a \in \mathcal{A}^R$ ): implica  $x_t \in R_a$  durante toda a duração do episódio.

Considerando novamente a solução apresentada na Figura 15, temos:

1. *Start-in* [**A**]: ocorre em  $\Pi_{a_1} = \{3\} \Rightarrow x_3 \in \mathbf{A}$
2. *Remain-in* [**B**]: em  $\Pi_{a_2} = \{5, 6, 7, 8\} \Rightarrow x_t \in \mathbf{B} \quad \forall (t \in \Pi_{a_2})$
3. *End-in* [**Goal**]: ocorre em  $\Pi_{a_3} = \{12\} \Rightarrow x_{12} \in \mathbf{B}$
4. *Remain-in* [**safe region**]: em  $\Pi_{a_4} = \{0 \dots T\} \Rightarrow x_t \notin \mathbf{D} \cup \mathbf{E} \quad \forall (t)$

## 5.3 Codificando o problema

Após estabelecido formalmente o problema, agora será definido o problema através de equações matemáticas detalhadas para cada um dos aspectos levantados. A seção 5.3.1 apresenta os aspectos principais que os autores em (ONO; WILLIAMS; BLACKMORE, 2013) codificaram, e que foram tomados como base neste trabalho. Em seguida, a seção 5.3.2 apresenta a codificação proposta nesta tese para o *scheduling* das missões.

### 5.3.1 Codificação previamente estabelecida na literatura

Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) estabelecem um modelo para o *CCQSP*, cujos aspectos mais importantes serão descritos a seguir.

$$\text{Minimizar } \mathcal{J}(\cdot) \quad (5.7)$$

sujeito a:

$$s \in S_F \quad (5.8)$$

$$\bar{x}_{t+1} = A\bar{x}_t + B\bar{u}_t \quad \forall(t) \quad (5.9)$$

$$\Pr \left[ \bigwedge_{a \in \Psi_c} \bigwedge_{t \in \Pi_a(s)} \mathbf{x}_t \in R_a \right] \geq 1 - \Delta_c \quad \forall(c) \quad (5.10)$$

A codificação apresentada é mais geral e serve para destacar os principais pontos codificados em (ONO; WILLIAMS; BLACKMORE, 2013). Uma função objetivo (5.7) deve ser minimizada, tipicamente considerando os controles esperados  $\bar{\mathbf{u}}_t$ , os estados esperados  $\bar{\mathbf{x}}_t$  e um *scheduling*  $s$  de tempo factível. A restrição (5.8) define  $s \in S_F$ , onde  $S_F$  representa o conjunto de todos os *scheduling* de tempos factíveis para o *CCQSP*. A restrição (5.9) é a versão determinística da equação (5.1) para a transição dos estados esperados  $\bar{\mathbf{x}}_t$  do veículo, dados os controles esperados  $\bar{\mathbf{u}}_t$ .

As *chance constraints* são descritas pela restrição (5.10), onde é considerado que tanto os controles  $\mathbf{u}_t$  quanto os estados  $\mathbf{x}_t$  são variáveis aleatórias como descrito na seção 5.2. Dessa forma, a restrição (5.10) estabelece que a probabilidade da variável aleatória  $\mathbf{x}_t$  pertencer ao subespaço  $R_a$  para cada episódio  $a \in \Psi_c$  e instantes  $t \in \Pi_a(s)$  deve ser maior ou igual a  $1 - \Delta_c$  para cada *chance constraint*. Uma codificação mais detalhada é introduzida em (ONO; WILLIAMS; BLACKMORE, 2013), onde as *chance-constraints* são estabelecidas. Por outro lado, uma codificação não é apresentada para a restrição (5.8), sendo implementado um algoritmo *branch-and-bound* para determinar o *scheduling*.

No presente trabalho, vamos partir da codificação já estabelecida em (ONO; WILLIAMS; BLACKMORE, 2013) para a restrição (5.10) e iremos introduzir uma codificação para a restrição (5.8). Assim, será possível pela primeira vez solucionar todo o problema utilizando um algoritmo de enumeração do tipo *branch-and-bound* através de *solvers* comerciais.

### 5.3.2 Codificando o scheduling

O *scheduling*  $s \in S_F$  é responsável por estabelecer os tempos em que cada evento ocorre, sendo utilizado pela restrição (5.10) para decidir os instantes  $t \in \Pi_a(s)$  em que teremos  $x_t \in R_a(s)$  em cada *chance-constraint*. Intuitivamente, codificar o *scheduling* é simplesmente marcar os instantes de tempo em que os eventos ocorrem e utilizar essa informação para determinar os instantes em que as restrições  $R_a$  serão obedecidas. Assim, considerando uma discretização  $\Delta t$  do horizonte de planejamento e definindo uma nova variável  $T_e \in \mathbb{R}^+$  para representar o instante de tempo em que o evento  $e$  ocorre, a equação (5.8) pode ser codificada utilizando as equações (5.11) e (5.12).

$$T_{e+1} = T_e + k_e \cdot \Delta t \quad \forall(e) \quad (5.11)$$

$$\mathcal{T}_r^{lb} \leq T_{e_r^E} - T_{e_r^S} \leq \mathcal{T}_r^{ub} \quad \forall(r \in \mathcal{T}) \quad (5.12)$$

A equação (5.11) supõe  $T_{e_0} = 0$  e  $k_e \in \mathbb{Z}_+$  como uma variável inteira positiva, onde  $k_e \cdot \Delta t$  determina o intervalo de tempo entre dois eventos consecutivos. O instante de tempo  $T_e$  exato em que cada evento ocorreu é determinado, mas sem garantir que o *scheduling* seja factível. Para isso, as restrições temporais  $\mathcal{T}$  são agora codificadas através da equação (5.12). Essa equação garante que o tempo decorrido ( $T_{e_r^E} - T_{e_r^S}$ ) entre o evento de início ( $e_r^S$ ) e fim ( $e_r^E$ ) esteja entre os limites inferior (*lb*) e superior (*ub*) estabelecidos para cada restrição temporal  $r \in \mathcal{T}$ . A informação a respeito dos *scheduling* factíveis será utilizada para definir os instantes  $t \in \Pi_a(s)$  em que  $\mathbf{x}_t \in R_a$ . Assim, deseja-se estabelecer uma forma de codificar o que está na equação (5.13).

$$\bigwedge_{t \in \Pi_a(s)} \mathbf{x}_t \in R_a \quad (5.13)$$

Para isso, definimos a variável  $Y_{at} \in \{0, 1\}$  tal que para  $Y_{at} = 1$  as restrições do episódio estão ativas  $\mathbf{x}_t \in R_a$ , permitindo reescrever a equação (5.13) usando a equação (5.14).

$$\bigwedge_t (Y_{at} \Rightarrow \mathbf{x}_t \in R_a) \quad (5.14)$$

O próximo passo é utilizar a codificação proposta para descrever os possíveis episódios. A definição dos episódios estabelece que sejam particionados em três tipos: Start-in ( $\mathcal{A}^S$ ), Remain-in ( $\mathcal{A}^R$ ) e End-in ( $\mathcal{A}^E$ ). Quando  $a \in \mathcal{A}^S$ , temos  $\mathbf{x}_t \in R_a$  no início do episódio o que estabelece  $Y_{at} = 1$  no instante  $t$  em que o episódio teve início ( $T_{e_a^S} = t \cdot \Delta t$ ) pela restrição (5.15). De forma semelhante, as restrições (5.16) e (5.17) estabelecem a validade da hipótese para  $\mathcal{A}^R$  e  $\mathcal{A}^E$ , respectivamente.

$$T_{e_a^S} = t \cdot \Delta t \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^S) \quad (5.15)$$

$$T_{e_a^S} \leq t \cdot \Delta t \leq T_{e_a^E} \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^R) \quad (5.16)$$

$$T_{e_a^E} = t \cdot \Delta t \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^E) \quad (5.17)$$

Uma vez determinados os *scheduling* factíveis e um meio para estabelecer a ativação das restrições dos episódios, a restrição (5.10) pode ser codificada segundo a restrição (5.18).

$$Pr \left[ \bigwedge_{a \in \Psi_c} \bigwedge_t (Y_{at} \Rightarrow \mathbf{x}_t \in R_a) \right] \geq 1 - \Delta_c \quad \forall (c) \quad (5.18)$$

O uso de implicações tautológicas ( $\Rightarrow$ ) nas restrições (5.14) - (5.18) pode ser modelado através de restrições lineares, conforme apresentado no capítulo 3. Resumindo, a codificação proposta para o *CCQSP* consiste em:

1. Minimizar uma função objetivo:  $\mathcal{J}(\cdot)$  (equação 5.7).
2. Manter sem mudanças o tratamento determinístico da equação de dinâmica (5.9).
3. Definir os instantes dos eventos ( $T_e$ ) e *scheduling* factíveis: restrições (5.11) e (5.12).
4. Armazenar em  $Y_{at}$  os instantes em que os episódios estão ativos: restrições (5.15)-(5.17).
5. Definir os riscos: restrição (5.18).

## 5.4 Decodificando o problema em restrições lineares

Com base na teoria matemática desenvolvida no capítulo 3 sobre variáveis lógicas, retomaremos o modelo estabelecido na seção 5.3.2 e apresentaremos uma forma eficiente de decodificá-lo como um modelo de Programação Linear Inteiro Misto (PLIM). As restrições (5.9), (5.11) e (5.12) já representam restrições lineares e podem ser codificadas como tal

diretamente. A restrição (5.15) (reescrita abaixo) pode ser codificada utilizando S7 da Tabela 6 do capítulo 3, onde para cada  $Y_{at} \in \mathbb{R}^+$  definiremos duas variáveis binárias auxiliares  $Y_{at}^1, Y_{at}^2 \in \{0, 1\}$  e cinco restrições, como apresentado na expressão (5.19).

$$T_{e_a^S} = t \cdot \Delta t \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^S) \quad (5.15)$$

$$\left\{ \begin{array}{l} T_{e_a^S} - t \cdot \Delta t \geq -MY_{at}^1 + \epsilon \\ T_{e_a^S} - t \cdot \Delta t \leq +MY_{at}^2 - \epsilon \\ Y_{at} \leq Y_{at}^1 \\ Y_{at} \leq Y_{at}^2 \\ Y_{at} \geq Y_{at}^1 + Y_{at}^2 - 1 \end{array} \right\} \quad \forall (t, a \in \mathcal{A}^S) \quad (5.19)$$

O mesmo pode ser feito para a restrição (5.17), trocando  $T_{e_a^S}$  e  $a \in \mathcal{A}^S$  na expressão (5.19) por  $T_{e_a^E}$  e  $a \in \mathcal{A}^E$ , respectivamente.

$$T_{e_a^E} = t \cdot \Delta t \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^E) \quad (5.17)$$

A restrição (5.16) utiliza S9 da Tabela 6 e resulta no conjunto de restrições da expressão (5.20). Também neste caso, para cada variável contínua  $Y_{at}$ , definiremos duas variáveis binárias auxiliares ( $Y_{at}^1, Y_{at}^2$ ) e cinco restrições.

$$T_{e_a^S} \leq t \cdot \Delta t \leq T_{e_a^E} \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^R) \quad (5.16)$$

$$\left\{ \begin{array}{l} T_{e_a^S} - t \cdot \Delta t \geq -MY_{at}^1 + \epsilon \\ T_{e_a^E} - t \cdot \Delta t \leq +MY_{at}^2 - \epsilon \\ Y_{at} \leq Y_{at}^1 \\ Y_{at} \leq Y_{at}^2 \\ Y_{at} \geq Y_{at}^1 + Y_{at}^2 - 1 \end{array} \right\} \quad \forall (t, a \in \mathcal{A}^R) \quad (5.20)$$

Por fim, é preciso decodificar a restrição (5.18).

$$Pr \left[ \bigwedge_{a \in \Psi_c} \bigwedge_t (Y_{at} \Rightarrow x_t \in R_a) \right] \geq 1 - \Delta_c \quad \forall (c) \quad (5.18)$$

Para isso, primeiro é preciso conhecer melhor o que é a região  $R_a$  e entendermos como podemos garantir, dentro do limite de probabilidade  $\Delta_c$ , que  $x_t \in R_a$  quando  $Y_{at} = 1$ . A região  $R_a$  é definida como um conjunto de  $j$  regiões que podem assumir dois tipos distintos:

1. **Ponto de passagem** ( $\mathbb{I}_j \mid j \in N_a^{\mathbb{I}}$ ): formado por um conjunto de  $k \in C_j^{\mathbb{I}}$  regiões convexas no qual  $x_t$  deve permanecer dentro de uma delas.

$$x_t \in \mathbb{I}_j \Leftrightarrow \bigvee_{k \in C_j^{\mathbb{I}}} \bigwedge_{i \in H_k^{\mathbb{I}}} h_i^T x_t \leq g_i \quad (5.21)$$

2. **Obstáculo** ( $\mathbb{O}_j \mid j \in N_a^{\mathbb{O}}$ ): formado por um conjunto de  $k \in C_j^{\mathbb{O}}$  regiões convexas no qual  $x_t$  deve permanecer fora de todas elas.

$$x_t \in \mathbb{O}_j \Leftrightarrow \bigwedge_{k \in C_j^{\mathbb{O}}} \bigvee_{i \in H_k^{\mathbb{O}}} h_i^T x_t \geq g_i \quad (5.22)$$

O termo  $h_i^T x_t = g_i$  é um hiperplano ( $i \in H$ ) no espaços do estados ( $\mathbb{R}^n$ ) que limita  $x_t$  a estar em um dos lados dependendo do tipo da região. Assim,  $R_a$  é formalmente definida pela restrição (5.23).

$$x_t \in R_a \Leftrightarrow \bigvee_{j \in N_a^{\mathbb{I}}} x_t \in \mathbb{I}_j \wedge \bigwedge_{j \in N_a^{\mathbb{O}}} x_t \in \mathbb{O}_j \quad (5.23)$$

A Figura 16 ilustra diferentes formas de modelar um único ponto de passagem (a) e um único obstáculo (b), respectivamente. Cada um deles pode ser modelado como uma região convexa simples, um conjunto unido de regiões convexas ou um conjunto sem interseção de regiões convexas. Na Figura 16c, é apresentado um exemplo de região  $R_a$  composta por dois pontos de passagem e dois obstáculos. Seguindo esse exemplo, a interpretação de  $x_t \in R_a$  será: no instante  $t$ ,  $x_t$  deve estar fora dos dois obstáculos definidos em cinza,  $\{1, 5\}$ , e dentro de somente um dos dois pontos de passagem em verde,  $\{2, 3, 4\}$ . Como exemplificado na Figura 16c, a região  $R_a$  pode ser otimizada ao remover o obstáculo 5, que é redundante. Afinal, uma vez que  $x_t$  deve estar dentro de um dos pontos de passagens, então ele já estará fora do obstáculo 5, pois a interseção de 5 com as regiões dos pontos de passagem é vazia. O obstáculo 1 também pode ser otimizado, mas nesse caso é possível apenas remover uma de suas arestas (linhas pontilhadas).

Pela definição 3, foi visto que esse problema é estocástico com  $\mathbf{x}_t$  e  $\mathbf{u}_t$  sendo variáveis aleatórias, onde é aplicado um controle em malha fechada, fazendo correções para que  $\mathbf{x}_t$  fique próximo de  $\bar{\mathbf{x}}_t$ . Isso é feito mediante a aplicação em tempo real de comandos extra nos controles para corrigir a trajetória e mitigar o crescimento da incerteza  $\Sigma_t$  (veja equações (5.3) e (5.4)). Nota-se que  $\mathbf{x}_t$  é variável aleatória por causa de distúrbios do ambiente e que  $\mathbf{u}_t$  se torna também variável aleatória por causa das correções aplicadas em tempo real. A definição 3 também estabeleceu que  $\mathbf{u}_t \in \mathbb{U}$  e  $\mathbf{x}_t \in \mathbb{X}$  pela equação (5.2), ou seja, não pode haver saturação nem dos controles nem dos estados. Entende-se por ocorrência de saturação o fato de  $\mathbf{u}_t \notin \mathbb{U}$  ou  $\mathbf{x}_t \notin \mathbb{X}$ . Assim, como  $\mathbf{x}_t$  e  $\mathbf{u}_t$  são variáveis aleatórias,

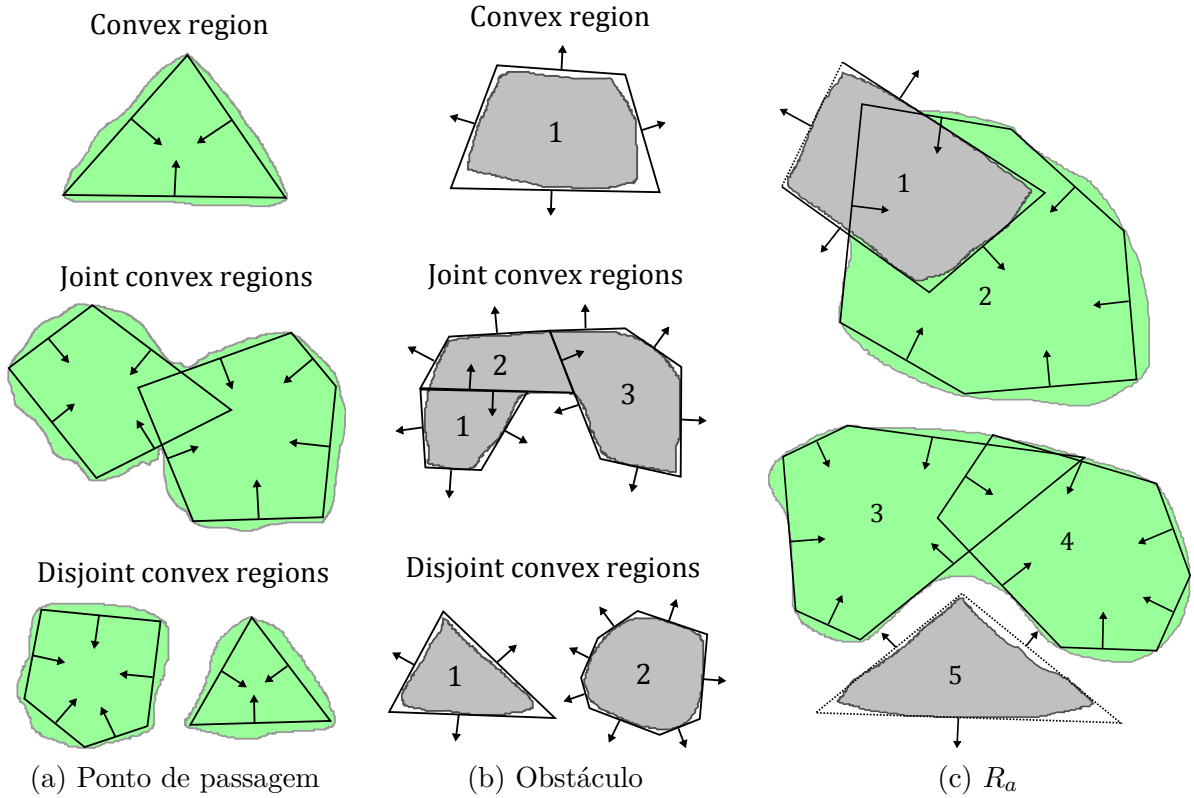


Figura 16 – Exemplos de regiões do tipo II, III e uma região  $R_a$  respectivamente

também é preciso que, dentro das margens de probabilidade  $\Delta_c$ , essas saturações não ocorram. Portanto, a restrição (5.18) será atualizada para a equação 5.24.

$$Pr \left[ \bigwedge_t \mathbf{u}_t \in \mathbb{U} \wedge \bigwedge_t \mathbf{x}_t \in \mathbb{X} \wedge \bigwedge_{a \in \Psi_c} \bigwedge_t (Y_{at} \Rightarrow x_t \in R_a) \right] \geq 1 - \Delta_c \quad \forall(c) \quad (5.24)$$

Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) resolvem a função de probabilidade  $Pr$  com a mesma ideia lógica apresentada na seção 4.3. Considerando, assim,  $\bar{\mathbf{x}}_t$  (valor esperado de  $\mathbf{x}_t$ ) e  $\bar{\mathbf{u}}_t$  (valor esperado de  $\mathbf{u}_t$ ) e limitando em  $\Delta_c$  a probabilidade da restrição não ser obedecida. Assim, com base no que os autores em (ONO; WILLIAMS; BLACKMORE, 2013) fizeram, a restrição (5.24) será decomposta nas restrições equivalentes (5.27) - (5.39). Neste caso, o risco de violar uma região qualquer é definido por  $m_{t,i}(\cdot)$  que é uma função contínua tal como definida na restrição (5.25). Essa função aloca o risco  $\xi$  de se estar fora de um hiperplano  $i$  qualquer.

$$m_{t,i}(\xi) = erf^{-1}(1 - 2\xi) \sqrt{2h_i^T \Sigma_t h_i} \quad (5.25)$$

O termo  $erf^{-1}$  é a função inversa do erro definida na restrição (5.26). Note que  $erf^{-1}$  é não linear, mas tomando o domínio  $0 < \xi \leq \Delta_c \leq 0.5$  a função  $erf^{-1}$  é estritamente

crescente, podendo, assim, ser aproximada como uma função linear por partes.

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (5.26)$$

Introduzidos esses conceitos, a seguir será indicado como codificar a restrição (5.24), onde basicamente é preciso considerar que:

- Não deve ocorrer a saturação dos controles ( $\mathbf{u}_t \in \mathbb{U}$ ).
- Não deve ocorrer a saturação dos estados ( $\mathbf{x}_t \in \mathbb{X}$ ).
- Restrições de ponto de passagem ( $Y_{at} \Rightarrow \bigvee_j \mathbf{x}_t \in \mathbb{I}_j$ ).
- Não colidir com obstáculos ( $Y_{at} \Rightarrow \bigwedge_j \mathbf{x}_t \in \mathbb{O}_j$ ).
- Garantir que a probabilidade de violação de qualquer um dos itens acima seja menor que o valor especificado ( $\Delta_c$ ).

#### Tratando o risco de saturação dos controles:

A restrição (5.27) deve manter  $\mathbf{u}_t$  dentro da região factível (controles não saturados) com o risco de ocorrer saturação dos controles, definido por  $m_{t,i}(\cdot)$ . O risco calculado pela função  $m_{t,i}(\cdot)$  é alocado na variável  $\delta_{it}^{\mathbb{U}}$ , considerando todos os instantes de tempo  $t$  e a possível violação de cada um dos hiperplanos ( $i \in H^{\mathbb{U}}$ ), que definem a região factível dos controles  $\mathbb{U}$ . A restrição (5.28) define o risco total ( $\delta^{\mathbb{U}}$ ) de haver saturação dos controles.

$$\bigwedge_{i \in H^{\mathbb{U}}} [g_i - h_i^T \bar{\mathbf{u}}_t \geq m_{t,i}(\delta_{it}^{\mathbb{U}})] \quad \forall(t) \quad (5.27)$$

$$\delta^{\mathbb{U}} = \sum_t \sum_{i \in H^{\mathbb{U}}} \delta_{it}^{\mathbb{U}} \quad (5.28)$$

#### Tratando o risco de saturação dos estados:

A restrição (5.29) mantém  $\mathbf{x}_t$  como estados não saturados de forma semelhante ao apresentado no item anterior. Dado os hiperplanos  $i \in H^{\mathbb{X}}$  da região factível dos estados  $\mathbb{X}$ , o risco de haver saturação é definido por  $m_{t,i}(\cdot)$  e alocado em  $\delta_{it}^{\mathbb{X}}$ . O risco total de haver saturação dos estados ( $\delta^{\mathbb{X}}$ ) está definido na restrição (5.30).

$$\bigwedge_{i \in H^{\mathbb{X}}} [g_i - h_i^T \bar{\mathbf{x}}_t \geq m_{t,i}(\delta_{it}^{\mathbb{X}})] \quad \forall(t) \quad (5.29)$$

$$\delta^{\mathbb{X}} = \sum_t \sum_{i \in H^{\mathbb{X}}} \delta_{it}^{\mathbb{X}} \quad (5.30)$$



**Tratando o risco de violar os pontos de passagem:**

A restrição (5.31) mantém  $\forall_j \mathbf{x}_t \in \mathbb{I}_j$  em todos os instantes  $t$  definidos pelo episódio ( $Y_{at} = 1$ ). A alocação do risco realizada pela função  $m_{t,i}(\cdot)$  é armazenada na variável  $\delta_{it}^{\mathbb{I}}$ . Assim, a restrição (5.32) define o risco total ( $\delta_c^{\mathbb{I}}$ ) de ocorrer a violação do ponto de passagem nos episódios associados a *chance constraint* ( $c$ ).

$$Y_{at} \Rightarrow \bigvee_{j \in N_a^{\mathbb{I}}} \bigvee_{k \in C_j^{\mathbb{I}}} \bigwedge_{i \in H_k^{\mathbb{I}}} \left[ g_i - h_i^T \bar{x}_t \geq m_{t,i}(\delta_{it}^{\mathbb{I}}) \right] \quad \forall(t, c, a \in \Psi_c) \quad (5.31)$$

$$\delta_c^{\mathbb{I}} = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^{\mathbb{I}}} \sum_{k \in C_j^{\mathbb{I}}} \sum_{i \in H_k^{\mathbb{I}}} \delta_{it}^{\mathbb{I}} \quad \forall(c) \quad (5.32)$$

A disjunção ( $\vee$ ) na restrição (5.31) pode ser decomposta no conjunto de relações da expressão 5.33, onde  $(Y_{at}^{\mathbb{I}j}, Y_{at}^{\mathbb{I}k}) \in \{0, 1\}$  são variáveis binárias auxiliares.

$$\left\{ \begin{array}{l} Y_{at} \Rightarrow \bigvee_{j \in N_a^{\mathbb{I}}} Y_{at}^{\mathbb{I}j} \\ Y_{at}^{\mathbb{I}j} \Rightarrow \bigvee_{k \in C_j^{\mathbb{I}}} Y_{at}^{\mathbb{I}k} \\ Y_{at}^{\mathbb{I}k} \Rightarrow \bigwedge_{i \in H_k^{\mathbb{I}}} \left[ g_i - h_i^T \bar{x}_t \geq m_{t,i}(\delta_{it}^{\mathbb{I}}) \right] \end{array} \right\} \quad \forall(t, c, a \in \Psi_c) \quad (5.33)$$

Agora, com base na teoria apresentada na seção 3, podemos aplicar *L5* (tautológica) e *G2* sobre as duas primeiras restrições e aplicar *C2* na terceira restrição. Neste caso, teremos um conjunto equivalente de restrições lineares, tal como apresentado nas restrições da expressão (5.34).

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{j \in N_a^{\mathbb{I}}} Y_{at}^{\mathbb{I}j} \\ Y_{at}^{\mathbb{I}j} \leq \sum_{k \in C_j^{\mathbb{I}}} Y_{at}^{\mathbb{I}k} \\ g_i - h_i^T \bar{x}_t \geq m_{t,i}(\delta_{it}^{\mathbb{I}}) + M(Y_{at}^{\mathbb{I}k} - 1) \end{array} \right\} \quad \forall(t, c, a \in \Psi_c) \quad (5.34)$$

Note que alguns conjuntos para os índices foram omitidos por falta de espaço nas equações das expressões (5.33) e (5.34). A relação dos índices omitidos para os conjuntos são:  $j \in N_a^{\mathbb{I}}$ ,  $k \in C_j^{\mathbb{I}}$  e  $i \in H_k^{\mathbb{I}}$ .

**Tratando o risco de colidir com obstáculos:**

A restrição (5.35) considera o risco de violação dos obstáculos ( $\bigwedge_j \mathbf{x}_t \in \mathbb{O}_j$ ) de forma semelhante ao definido anteriormente para os pontos de passagem. A diferença é que precisamos estar fora de todos os obstáculos. Para isso, deve-se estar fora de apenas um dos hiperplanos ( $i \in H_k^{\mathbb{O}}$ ) que definem o obstáculo. A restrição (5.36) acumula o risco total ( $\delta_c^{\mathbb{O}}$ ) de haver colisão com os obstáculos associados aos episódios  $a \in \Psi_c$ .

$$Y_{at} \Rightarrow \bigvee_{i \in H_k^{\mathbb{O}}} \left[ h_i^T \bar{x}_t - g_i \geq m_{t,i}(\delta_{kt}^{\mathbb{O}}) \right] \quad \forall(t, c, a \in \Psi_c, j \in N_a^{\mathbb{O}}, k \in C_j^{\mathbb{O}}) \quad (5.35)$$

$$\delta_c^{\text{O}} = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^{\text{O}}} \sum_{k \in C_j^{\text{O}}} \delta_{kt}^{\text{O}} \quad \forall(c) \quad (5.36)$$

A disjunção (V) na restrição (5.35) também pode ser tratada de forma similar ao realizado anteriormente para os pontos de passagem. Assim, ao decompor a expressão (5.35), teremos a expressão (5.37), onde  $Y_{at}^{\text{O}i} \in \{0, 1\}$  representa variáveis binárias auxiliares.

$$\left\{ \begin{array}{l} Y_{at} \Rightarrow \bigvee_{i \in H_k^{\text{O}}} Y_{at}^{\text{O}i} \\ Y_{at}^{\text{O}i} \Rightarrow [h_i^T \bar{x}_t - g_i \geq m_{t,i}(\delta_{kt}^{\text{O}})] \end{array} \right\} \quad \forall(t, c, a \in \Psi_c, j \in N_a^{\text{O}}, k \in C_j^{\text{O}}) \quad (5.37)$$

Também poderemos aplicar  $L5$  (tautológica) e  $G2$  sobre a primeira equação e aplicar  $C2$  na segunda equação para obtermos as restrições lineares equivalentes na expressão (5.38).

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{i \in H_k^{\text{O}}} Y_{at}^{\text{O}i} \\ h_i^T \bar{x}_t - g_i \geq m_{t,i}(\delta_{kt}^{\text{O}}) + M(Y_{at}^{\text{O}i} - 1) \end{array} \right\} \quad \forall(t, c, a \in \Psi_c, j \in N_a^{\text{O}}, k \in C_j^{\text{O}}) \quad (5.38)$$

#### Limitando a alocação do risco:

Por fim, a restrição (5.39) limita todos os riscos alocados pelo modelo ao valor  $\Delta_c$  especificado no CCQSP.

$$\delta^{\text{U}} + \delta^{\text{X}} + \delta_c^{\text{I}} + \delta_c^{\text{O}} \leq \Delta_c \quad \forall(c) \quad (5.39)$$

### 5.4.1 Modelo completo do CCQSP Planning Problem

O modelo completo para resolver o CCQSP é descrito a seguir:

$$\text{Minimizar } \mathcal{J}(\cdot) \quad (5.7)$$

Equação de dinâmica:

$$\bar{x}_{t+1} = A\bar{x}_t + B\bar{u}_t \quad \forall(t) \quad (5.9)$$

Instante em que ocorre os eventos (todos os *scheduling*):

$$T_{e+1} = T_e + k_e \cdot \Delta t \quad \forall(e) \quad (5.11)$$

Restrições temporais (apenas *scheduling* factíveis):

$$\mathcal{T}_r^{lb} \leq T_{e_r^E} - T_{e_r^S} \leq \mathcal{T}_r^{ub} \quad \forall (r \in \mathcal{T}) \quad (5.12)$$

Episódios Start-In:

$$\left\{ \begin{array}{l} T_{e_a^S} - t \cdot \Delta t \geq -MY_{at}^1 + \epsilon \\ T_{e_a^S} - t \cdot \Delta t \leq +MY_{at}^2 - \epsilon \\ Y_{at} \leq Y_{at}^1 \\ Y_{at} \leq Y_{at}^2 \\ Y_{at} \geq Y_{at}^1 + Y_{at}^2 - 1 \end{array} \right\} \quad \forall (t, a \in \mathcal{A}^S) \quad (5.19)$$

Episódios End-In:

$$\left\{ \begin{array}{l} T_{e_a^E} - t \cdot \Delta t \geq -MY_{at}^1 + \epsilon \\ T_{e_a^E} - t \cdot \Delta t \leq +MY_{at}^2 - \epsilon \\ Y_{at} \leq Y_{at}^1 \\ Y_{at} \leq Y_{at}^2 \\ Y_{at} \geq Y_{at}^1 + Y_{at}^2 - 1 \end{array} \right\} \quad \forall (t, a \in \mathcal{A}^E) \quad (5.40)$$

Episódios Remain-In:

$$\left\{ \begin{array}{l} T_{e_a^S} - t \cdot \Delta t \geq -MY_{at}^1 + \epsilon \\ T_{e_a^E} - t \cdot \Delta t \leq +MY_{at}^2 - \epsilon \\ Y_{at} \leq Y_{at}^1 \\ Y_{at} \leq Y_{at}^2 \\ Y_{at} \geq Y_{at}^1 + Y_{at}^2 - 1 \end{array} \right\} \quad \forall (t, a \in \mathcal{A}^R) \quad (5.20)$$

Alocando o risco de saturação dos controles:

$$g_i - h_i^T \bar{\mathbf{u}}_t \geq m_{t,i}(\delta_{it}^U) \quad \forall (t, i \in H^U) \quad (5.41)$$

$$\delta^U = \sum_t \sum_{i \in H^U} \delta_{it}^U \quad (5.28)$$

Alocando o risco de saturação dos estados:

$$g_i - h_i^T \bar{\mathbf{x}}_t \geq m_{t,i}(\delta_{it}^X) \quad \forall (t, i \in H^X) \quad (5.42)$$

$$\delta^X = \sum_t \sum_{i \in H^X} \delta_{it}^X \quad (5.30)$$

Alocando o risco de violar os pontos de passagem:

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{j \in N_a^I} Y_{at}^{Ij} \\ Y_{at}^{Ij} \leq \sum_{k \in C_j^I} Y_{at}^{Ik} \\ g_i - h_i^T \bar{x}_t \geq m_{t,i}(\delta_{it}^I) + M(Y_{at}^{Ik} - 1) \end{array} \right\} \quad \forall (t, c, a \in \Psi_c) \quad (5.34)$$

$$\delta_c^I = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^I} \sum_{k \in C_j^I} \sum_{i \in H_k^I} \delta_{it}^I \quad \forall (c) \quad (5.32)$$

Alocando o risco de colidir com obstáculos:

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{i \in H_k^O} Y_{at}^{Oi} \\ h_i^T \bar{x}_t - g_i \geq m_{t,i}(\delta_{kt}^O) + M(Y_{at}^{Oi} - 1) \end{array} \right\} \quad \forall (t, c, a \in \Psi_c, j \in N_a^O, k \in C_j^O) \quad (5.38)$$

$$\delta_c^O = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^O} \sum_{k \in C_j^O} \delta_{kt}^O \quad \forall (c) \quad (5.36)$$

Limitando a alocação do risco:

$$\delta^U + \delta^X + \delta_c^I + \delta_c^O \leq \Delta_c \quad \forall (c) \quad (5.39)$$

Limitantes das variáveis:

$$\bar{\mathbf{x}}_t \in \mathbb{R}^n, \bar{\mathbf{u}}_t \in \mathbb{R}^m \quad (5.43)$$

$$(T_e, \delta^U, \delta^X, \delta_c^I, \delta_c^O, \delta_{it}^U, \delta_{it}^X, \delta_{it}^I, \delta_{kt}^O) \geq 0 \quad (5.44)$$

$$k_e \in \mathbb{Z}_+ \quad (5.45)$$

$$0 \leq (Y_{at}) \leq 1 \quad (5.46)$$

$$(Y_{at}^1, Y_{at}^2, Y_{at}^{Ij}, Y_{at}^{Oi}) \in \{0, 1\} \quad (5.47)$$

## 5.5 Melhorias do modelo

### 5.5.1 Equações de dinâmica com efeito de arrasto

Os trabalhos descritos em (LI, 2010; ONO; WILLIAMS; BLACKMORE, 2013) consideram uma planta como mostrada nas equações 5.48-5.49.

$$A = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} \Delta t^2/2 & 0 \\ 0 & \Delta t^2/2 \\ \Delta t & 0 \\ 0 & \Delta t \end{pmatrix} \quad (5.48)$$

$$\|\mathbf{u}_t\| \leq a^{max}, \quad \|C\mathbf{x}_t\| \leq v^{max} \quad (5.49)$$

onde

$$C = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Logo, a equação que representa a dinâmica do VANT não considera forças externas como a resistência do ar. Isso é ruim, pois permite que, uma vez alcançada a velocidade desejada, o VANT consiga manter o voo sem aplicação de controles adicionais, valendo-se apenas da inércia para manter sua velocidade e a direção. Trata-se de um modelo de dinâmica pouco realista para se estimar o custo da rota, pois prevê que não há consumo de combustível para manter a velocidade e direção.

O que vamos propor nesta seção é um modelo para auxiliar numa melhor tomada de decisão, estabelecendo uma nova planta que incorpora de forma aproximada os efeitos da resistência do ar sobre o VANT.

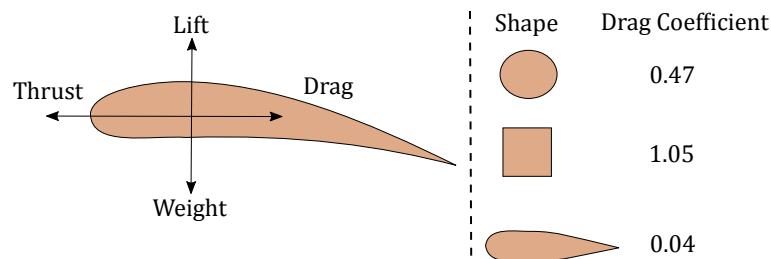


Figura 17 – Forças aerodinâmicas e coeficientes de arrasto

A Figura 17 apresenta as quatro principais forças aerodinâmicas atuantes num VANT. Para um voo num plano 2D, onde é considerado que a aeronave esteja voando sempre a uma altura constante, podemos ignorar as forças *Weight*, causada pelo efeito da gravidade, e *Lift* que provê a sustentação da aeronave. Assim, restam as forças *Thrust*

( $F^T$ ), que é o impulso causado pelo motor do VANT, e *Drag* ( $F^D$ ), causada pela resistência do ar.

Logo, podemos considerar que a força resultante ( $F^R$ ) do sistema é dada pela equação (5.50). Estabelecendo uma comparação com a literatura, a proposta de (ONO; WILLIAMS; BLACKMORE, 2013) considerava apenas que  $F^R = F^T$  nos experimentos realizados.

$$F^R = F^T - F^D \quad (5.50)$$

A força de resistência do ar é calculada como descrito na equação (5.51), em que  $\rho$  é a densidade do ar,  $v$  é a velocidade do VANT relativo ao ar,  $C_d$  (*drag coefficient*) é um número adimensional referente ao formato das superfícies aerodinâmicas do VANT. A Figura 17 apresenta exemplos de  $C_d$  para algumas formas. Por fim,  $A$  é a área da seção transversal do VANT (*cross sectional area*).

$$F^D = \frac{1}{2}\rho v^2 C_d A \quad (5.51)$$

As forças  $F^T$  e  $F^D$  sofrem variações ao longo do tempo, mas podemos considerar que permanecem constantes durante o intervalo discretizado em  $\Delta t$ . Isso, claro, se pensarmos em pequenos valores de  $\Delta t$ . Assim, podemos deduzir entre cada instante  $t$  e  $t + 1$  uma aceleração resultante constante para o VANT definida por:

$$a_t^R = a_t - K v_t^2, \quad (5.52)$$

onde  $K$  é o termo constante de  $F^D$  dividido pela massa do VANT como segue:

$$K = \frac{1}{2m}\rho C_d A. \quad (5.53)$$

Com uma aceleração resultante constante  $a_t^R$ , podemos estabelecer as equações de transição discreta para a velocidade e posição como:

$$v_{t+1} = v_t + a_t^R \Delta t \quad (5.54)$$

$$p_{t+1} = p_t + v_t \Delta t + a_t^R \Delta t^2 / 2 \quad (5.55)$$

Substituindo (5.52) em (5.54) e (5.55) temos:

$$v_{t+1} = v_t + [a_t - K v_t^2] \Delta t \quad (5.56)$$

$$p_{t+1} = p_t + v_t \Delta t + [a_t - K v_t^2] \Delta t^2 / 2 \quad (5.57)$$

Entretanto, há o termo quadrático da velocidade  $v_t^2$  nas equações (5.56) e (5.57), o que inviabilizará a construção de um modelo de PLIM para o problema, pois  $v_t$  é uma variável do modelo. Por isso, vamos aproximar  $v_t^2$  por  $v_t v^{term}$ , onde  $v^{term}$  (ou  $v^{max}$ ) é a velocidade terminal do VANT, e teremos as equações lineares (5.58) e (5.59).

$$v_{t+1} = v_t [1 - K v^{term} \Delta t] + a_t \Delta t \quad (5.58)$$

$$p_{t+1} = p_t + v_t [\Delta t - K v^{term} \Delta t^2 / 2] + a_t \Delta t^2 / 2 \quad (5.59)$$

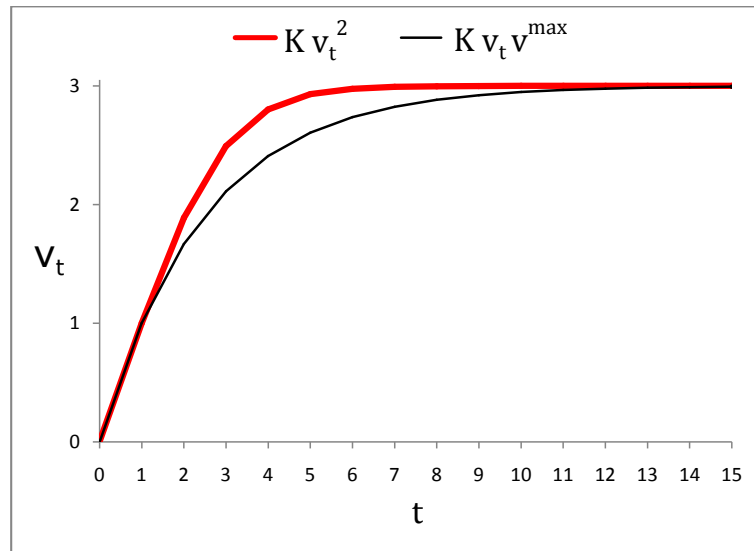


Figura 18 – Modelo conservativo da resistência do ar

Dessa forma, temos que as equações (5.58) e (5.59) são conservativas em relação as equações (5.56) e (5.57), uma vez que estabelecem um pior caso para a força de resistência do ar. A Figura 18 apresenta um exemplo para ambos os casos, onde é aplicada uma aceleração constante ao VANT de  $u_{max} = 1$ , que estava inicialmente em repouso ( $t = 0$ ,  $v_0 = 0$ ). Nota-se que ambas as equações levam o VANT a uma mesma velocidade terminal de  $v^{term} = 3m/s$  e, como esperado, as equações (5.58) e (5.59), por serem conservadoras, levam um tempo maior para alcançar o mesmo estado.

As matrizes considerando a resistência do ar são:

$$A = \begin{pmatrix} 1 & 0 & K_p & 0 \\ 0 & 1 & 0 & K_p \\ 0 & 0 & K_v & 0 \\ 0 & 0 & 0 & K_v \end{pmatrix}, \quad B = \begin{pmatrix} \Delta t^2 / 2 & 0 \\ 0 & \Delta t^2 / 2 \\ \Delta t & 0 \\ 0 & \Delta t \end{pmatrix} \quad (5.60)$$

$$\|\mathbf{u}_t\| \leq a^{max}, \quad \|C\bar{\mathbf{x}}_t\| \leq v^{max} \quad (5.49)$$

onde:

$$K_p = \Delta t - K v^{term} \Delta t^2 / 2 \quad (5.61)$$

$$K_v = 1 - K v^{term} \Delta t \quad (5.62)$$

Para utilizarmos a nova dinâmica dada pela expressão (5.60), precisamos conhecer a constante  $K$  para o VANT a ser modelado. Vimos pelas equações (5.53) e (5.51) que a constante depende de vários fatores: massa do VANT ( $m$ ), densidade do ar ( $\rho$ ), *drag coefficient* ( $C_d$ ) and *cross sectional area*  $A$ .

Todavia, o valor de  $K$  pode ser encontrado de outra forma. Observe na Figura 18 que, quando aplicamos a aceleração máxima  $a^{max}$  num sistema com resistência do ar, uma velocidade terminal  $v^{term}$  se apresentará após algum tempo  $t = 0, 1, \dots, \tau$ . Podemos estabelecer assim que  $v_\tau = v^{term}$  e sabemos que quando o VANT alcançar a velocidade terminal teremos uma aceleração resultante nula do sistema e uma aceleração máxima dos controles  $a_\tau = a^{max}$ . A equação (5.52) pode ser reescrita e a constante  $K$  pode ser calculada diretamente pela equação (5.63).

$$0 = a_\tau - K v_\tau^2 \quad \Leftrightarrow \quad K = \frac{a^{max}}{(v^{term})^2} \quad (5.63)$$

Para garantir a aplicabilidade da nova planta (5.60), deve-se garantir que  $K_p > 0$  e  $K_v > 0$ . Isso pode ser feito ao escolher uma discretização do tempo  $\Delta t$  menor que:

$$\Delta t < \frac{1}{K v^{term}} \quad \Leftrightarrow \quad \Delta t < \frac{v^{term}}{a^{max}} \quad (5.64)$$

### 5.5.2 Restrições de risco reduzidas

Nesta seção será proposta uma forma mais enxuta de codificar as *Chance Constraints*. Vimos na seção 5.4 que para fazer a alocação dos riscos das *Chance Constraints* eram utilizadas as variáveis  $\delta_{it}^U \geq 0$ ,  $\delta_{it}^X \geq 0$ ,  $\delta_{it}^I \geq 0$  e  $\delta_{kt}^O \geq 0$  na alocação do risco  $\xi$ . A alocação do risco era calculada pelas equações (5.25) e (5.26):

$$m_{t,i}(\xi) = erf^{-1}(1 - 2\xi) \sqrt{2h_i^T \Sigma_t h_i} \quad (5.25)$$

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (5.26)$$



onde as variáveis  $\delta_{it}^{\mathbb{U}}$ ,  $\delta_{it}^{\mathbb{X}}$  e  $\delta_{it}^{\mathbb{I}}$  estão relacionadas a cada hiperplano que define as regiões internas de  $\mathbf{u}_t \in \mathbb{U}$ ,  $\mathbf{x}_t \in \mathbb{X}$  e  $\mathbf{x}_t \in \mathbb{I}$ . A variável  $\delta_{kt}^{\mathbb{O}}$  está relacionada a cada obstáculo convexo  $k$ .

O problema nesta abordagem está na soma excessiva dos riscos como apresentado nas equações (5.28), (5.30), (5.32), (5.36) e (5.39). O que torna o modelo muito conservador como reportado em (ONO; WILLIAMS; BLACKMORE, 2013).

$$\delta^{\mathbb{U}} = \sum_t \sum_{i \in H^{\mathbb{U}}} \delta_{it}^{\mathbb{U}} \quad (5.28)$$

$$\delta^{\mathbb{X}} = \sum_t \sum_{i \in H^{\mathbb{X}}} \delta_{it}^{\mathbb{X}} \quad (5.30)$$

$$\delta_c^{\mathbb{I}} = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^{\mathbb{I}}} \sum_{k \in C_j^{\mathbb{I}}} \sum_{i \in H_k^{\mathbb{I}}} \delta_{it}^{\mathbb{I}} \quad \forall(c) \quad (5.32)$$

$$\delta_c^{\mathbb{O}} = \sum_t \sum_{a \in \Psi_c} \sum_{j \in N_a^{\mathbb{O}}} \sum_{k \in C_j^{\mathbb{O}}} \delta_{kt}^{\mathbb{O}} \quad \forall(c) \quad (5.36)$$

$$\delta^{\mathbb{U}} + \delta^{\mathbb{X}} + \delta_c^{\mathbb{I}} + \delta_c^{\mathbb{O}} \leq \Delta_c \quad \forall(c) \quad (5.39)$$

Se analisarmos bem a natureza das incertezas inerentes ao VANT, podemos perceber que há 3 tipos de incerteza relativas às variáveis de controle, variáveis de velocidade e variáveis de posição. Tais incertezas estão em torno dos valores esperados de  $\bar{\mathbf{u}}_t$ ,  $\bar{\mathbf{v}}_t$  e  $\bar{\mathbf{p}}_t$  como apresentado na Figura 19.

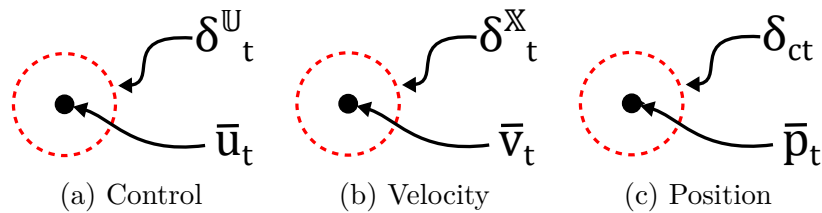


Figura 19 – Incertezas nas variáveis de controle  $\delta_t^{\mathbb{U}}$ , velocidade  $\delta_t^{\mathbb{X}}$  e posição  $\delta_{ct}$

Essa incerteza define um raio de segurança que é independente dos hiperplanos, mas dependente apenas do número de dimensões para representar a posição do veículo. Os hiperplanos que definem separações no espaço dos controles e estados são necessários apenas para calcular o raio de segurança na alocação do risco. Todavia, nota-se que um raio de segurança menor resulta em um risco maior e contém todos os riscos menores. Por exemplo, para uma variável aleatória  $x \sim \mathcal{N}(0, \sigma)$ , há um risco 2.5% de  $x > 2\sigma$  e um risco de 0.15% de  $x > 3\sigma$ . Esse exemplo deixa claro que o risco de 0.15% está contido no risco de 2.5%, pois se  $x > 3 \Rightarrow x > 2$ . O risco de um raio de segurança maior sempre está

contido dentro do risco de um raio menor. Logo, é possível garantir a segurança sem a necessidade de somar os riscos menores alocados (raios maiores), mas sim apenas o risco calculado para o menor raio de segurança e considerando uma distribuição multivariada para tratar adequadamente o número de dimensões. Assim, assume-se o maior risco para a variável que irá conter todos os demais riscos. A Figura 20 apresenta as variáveis e os raios associados ao cálculo do maior risco, para as regiões que definem: saturação dos controles  $\mathbb{U}$ , saturação dos estados  $\mathbb{X}$ , pontos de passagem  $\mathbb{I}$  e obstáculos  $\mathbb{O}$ .

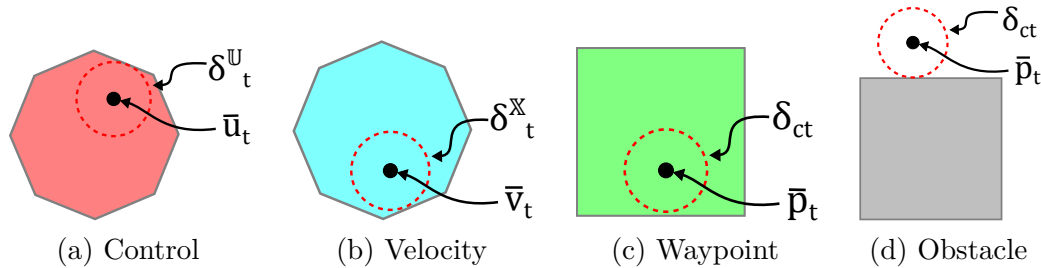


Figura 20 – Raio de segurança na alocação do risco para:  $\mathbb{U}$ ,  $\mathbb{X}$ ,  $\mathbb{I}$  e  $\mathbb{O}$

Portanto, uma modelagem mais enxuta consiste em substituímos as variáveis anteriores  $\delta_{it}^{\mathbb{U}}$ ,  $\delta_{it}^{\mathbb{X}}$ ,  $\delta_{it}^{\mathbb{I}}$  e  $\delta_{kt}^{\mathbb{O}}$  pelas variáveis  $\delta_t^{\mathbb{U}}$ ,  $\delta_t^{\mathbb{X}}$  e  $\delta_{ct}$ . As equações (5.28), (5.30), (5.32), (5.36) e (5.39) serão substituídas pela equação abaixo. As equações 5.25 e 5.26 são tomadas com base numa análise multivariada.

$$\sum_t \delta_t^{\mathbb{U}} + \sum_t \delta_t^{\mathbb{X}} + \sum_t \delta_{ct} \leq \Delta_c \quad \forall(c) \quad (5.65)$$

### 5.5.3 Adicionando restrições para fortalecer o modelo

Nesta seção será apresentado como fortalecer o modelo proposto ao adicionar novas restrições válidas para acelerar sua resolução. Vamos considerar as situações a seguir:

**Caso 1 :** O episódio estará ativo em algum instante.

**Caso 2 :** Existem episódios do tipo Remain-In que estarão ativos sempre.

**Caso 3 :** Regiões de um ponto de passagem excluem-se mutuamente.

**Caso 4 :** Hiperplanos de um obstáculo excluem-se mutuamente.

#### 5.5.3.1 O episódio estará ativo em algum instante ( Caso 1 )

Primeiramente, serão considerados apenas os episódios do tipo Start-In e End-In. O fortalecimento proposto consiste em adicionar a restrição 5.66 ao modelo. Ao fazer isso,

o problema continuará com o mesmo espaço de busca factível, mas estaremos melhorando o limitante inferior do problema (*lower-bound*).

$$\sum_t Y_{at} = 1 \quad \forall (a \in \mathcal{A}^{S|E}) \quad (5.66)$$

A Figura 21 ilustra um exemplo do que acontece quando as variáveis de marcação do tempo são relaxadas ( $k_e \in \mathbb{R}$ ). Duas missões distintas foram estabelecidas no CCQSP, definindo a ordem nos pontos de passagem como: Mission 1 percorre  $Start \rightarrow A \rightarrow B \rightarrow Goal$ , Mission 2 percorre  $Start \rightarrow B \rightarrow A \rightarrow Goal$ . Ambas as missões também especificam que o tempo em  $A \Leftrightarrow B$  deve ser menor que 5 unidades de  $\Delta t$ .

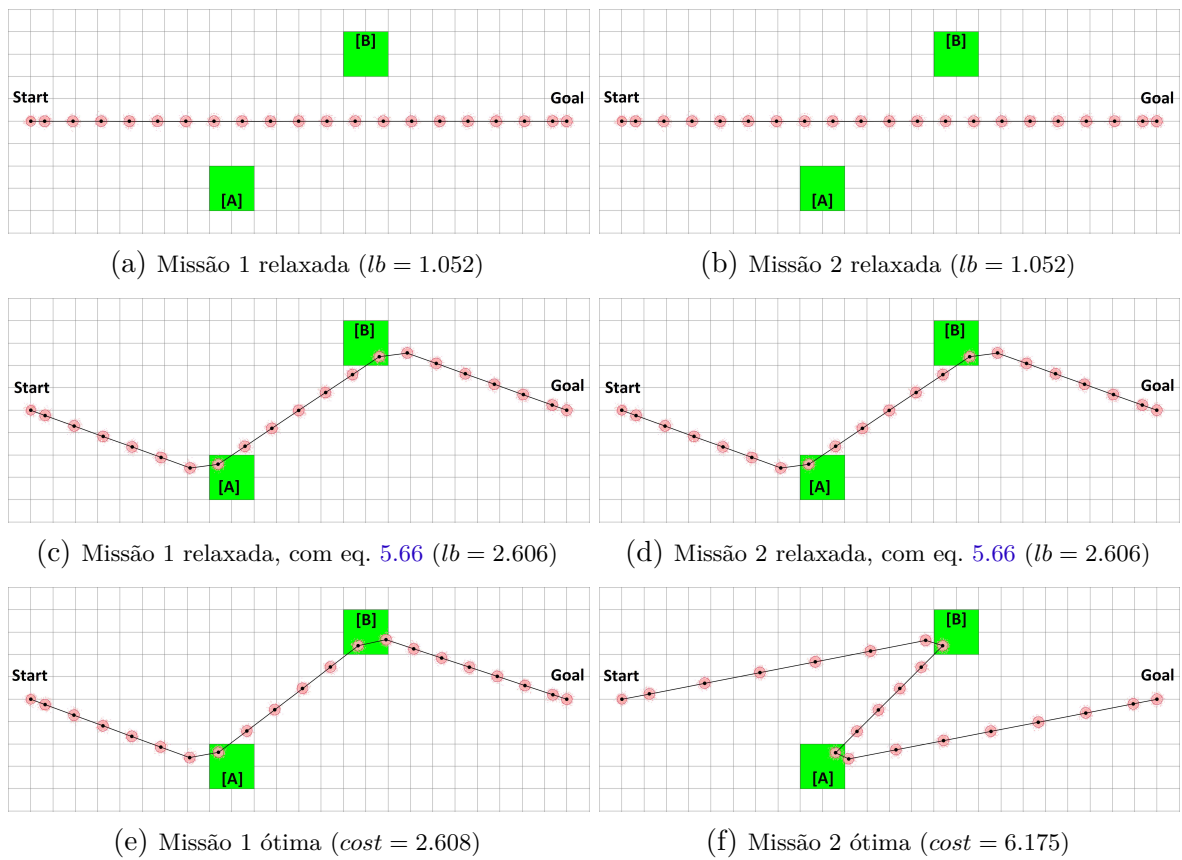


Figura 21 – Exemplos de limitantes para o CCQSP para duas missões diferentes

As imagens do lado esquerdo na Figura 21 representam a Missão 1 e ao lado direito a Missão 2. As Figuras 21a e 21b mostram o que acontece quando a variável de marcação de tempo é relaxada ( $k_e \in \mathbb{R}$ ). Nesses casos, as soluções obtidas ignoram completamente os locais de passagem definidos na missão, além de fornecerem limitantes inferiores (*lower bound* or  $lb$ ) bastante ruins (0.052). Isso ocorre por causa do tempo  $T_e$  não assumir mais valores múltiplos de  $\Delta t$ , assim as equações 5.15 e 5.17 não levam à ativação dos episódios definidos pela missão. As Figuras 21c e 21d apresentam o resultado com a mesma relaxação, mas com a adição do fortalecimento proposto pela equação 5.66. Note que o veículo foi

forçado a passar pelos locais indicados na missão e as soluções fornecem agora um melhor limitante inferior (2.606). Todavia, ambas as soluções continuam relaxadas e não são factíveis ainda como solução do problema. A solução na Figura 21c mantém a ordem correta na passagem dos locais especificados na missão, mas viola a factibilidade de manter as 5 unidades de  $\Delta t$  especificadas na missão (note que a rota relaxada utiliza 6 unidades). A solução na Figura 21d viola ambas as situações. Finalmente, as Figuras 21e e 21f ilustram a solução ótima para o CCQSP em cada missão. A solução ótima é obtida restaurando  $k_e \in \mathbb{Z}$  e se trata da mesma solução obtida ao resolver o problema com e sem a equação 5.66. A aplicação da restrição 5.66 define uma restrição de fortalecimento válida porque força a ativação dos episódios ao menos uma vez, o que já é esperado na solução ótima do problema. Isso melhora o limitante inferior do modelo ao permitir ao método Branch & Cut fazer mais cortes durante sua execução, podendo levar a uma redução no tempo computacional para solucionar o problema.

Considerando o mesmo princípio para os episódios do tipo Remain-In, será adicionada a restrição 5.67 ao modelo. Essa nova equação estabelece que o número de ativações deve estar entre um número mínimo e máximo. Os valores mínimo e máximo do episódio são obtidos através dos limitantes da respectiva restrição temporal ( $r \in \mathcal{T}$ ), que é requisito necessário ao cumprimento da missão.

$$\left\lceil \frac{\mathcal{T}_r^{lb}}{\Delta t} \right\rceil \leq \sum_t Y_{at} \leq \left\lfloor \frac{\mathcal{T}_r^{ub}}{\Delta t} \right\rfloor \quad \forall (a \in \mathcal{A}^R, r \in \mathcal{T} | e_a^S = e_r^S \wedge e_a^E = e_r^E) \quad (5.67)$$

### 5.5.3.2 *Existem episódios Remain-In que estarão ativos sempre ( Caso 2 )*

Se existirem episódios do tipo Remain-In que ligam o primeiro evento ( $e_0$ ) ao último ( $e_E$ ), então estes podem ser otimizados ao fixar  $Y_{at} = 1$  e remover esses episódios da restrição 5.16. A equação 5.68 estabelece a fixação deles e a equação 5.69 apresenta a atualização da equação 5.16, após remove-los. Esta situação ocorre com frequência ao estabelecer uma navegação segura, onde desejamos que o veículo esteja fora de todos os obstáculos do início ao fim da missão, isso pode ser visto no episódio  $a_4$  na Figura 15 onde é previsto que ele estará ativo entre  $e_0$  e  $e_E$ .

$$Y_{at} = 1 \quad \forall (t, a \in \mathcal{A}^R | e_a^S = e_0 \wedge e_a^E = e_E) \quad (5.68)$$

$$T_{e_a^S} \leq t \cdot \Delta t \leq T_{e_a^E} \quad \Rightarrow \quad Y_{at} \quad \forall (t, a \in \mathcal{A}^R | e_a^S \neq e_0 \vee e_a^E \neq e_E) \quad (5.69)$$

### 5.5.3.3 *Regiões de um ponto de passagem excluem-se mutuamente ( Caso 3 )*

Quando um ponto de passagem é estabelecido contendo duas ou mais regiões convexas ( $k \in C^{\mathbb{I}}$ ), a ativação deverá ocorrer para pelo menos uma das regiões. Isso foi

descrito formalmente pela equação 5.34, cujos aspectos mais relevantes estão replicados na equação 5.70. Todavia, como a ativação de apenas uma das regiões convexas é condição suficiente, pode-se considerar que as regiões se excluem mutuamente e a equação 5.70 pode ser substituída pela equação 5.71. Essa equação estabelece a exclusão mútua por meio da igualdade. Além disso, podemos transformar a variável binária  $Y_{at}^{\mathbb{I}j}$  em variável contínua, mantendo  $Y_{at}^{\mathbb{I}k}$  como binária visto que  $Y_{at}^{\mathbb{I}j}$  é definida em função de  $Y_{at}^{\mathbb{I}k}$ . Assim, as equações fortalecidas ficam como descrito na equação 5.71.

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{j \in N_a^{\mathbb{I}}} Y_{at}^{\mathbb{I}j} \\ Y_{at}^{\mathbb{I}j} \leq \sum_{k \in C_j^{\mathbb{I}}} Y_{at}^{\mathbb{I}k} \end{array} \quad \forall (j \in N_a^{\mathbb{I}}) \right\} \quad \forall (t, a) \quad (5.70)$$

$$\left\{ \begin{array}{l} Y_{at} = \sum_{j \in N_a^{\mathbb{I}}} Y_{at}^{\mathbb{I}j} \\ Y_{at}^{\mathbb{I}j} = \sum_{k \in C_j^{\mathbb{I}}} Y_{at}^{\mathbb{I}k} \end{array} \quad \forall (j \in N_a^{\mathbb{I}}) \right\} \quad \forall (t, a) \quad (5.71)$$

#### 5.5.3.4 Hiperplanos de um obstáculo excluem-se mutuamente ( Caso 4 )

A equação 5.38 definiu que se o episódio está ativo ( $Y_{at} = 1$ ), então o VANT deve estar fora de pelo menos um dos hiperplanos de cada região convexa ( $k \in C^{\mathbb{O}}$ ) que define o obstáculo. Essa lógica é descrita novamente na equação 5.72. Como a ativação de apenas um hiperplano é condição suficiente para garantir que o VANT esteja fora da região convexa, então podemos fortalecer a equação 5.72 definindo a exclusão mútua dos hiperplanos e estabelecendo a igualdade, onde teremos a equação 5.73

$$Y_{at} \leq \sum_{i \in H_k^{\mathbb{O}}} Y_{at}^{\mathbb{O}i} \quad \forall (t, a, k \in C^{\mathbb{O}}) \quad (5.72)$$

$$Y_{at} = \sum_{i \in H_k^{\mathbb{O}}} Y_{at}^{\mathbb{O}i} \quad \forall (t, a, k \in C^{\mathbb{O}}) \quad (5.73)$$

## 5.6 Análise de complexidade

Nesta seção estudos a respeito da complexidade do CCQSP serão feitos a fim de definir seus limites de aplicação e buscar maior aprofundamento matemático a respeito do CCQSP. Esse aprofundamento pode tornar possível a elaboração de soluções que buscam contornar o problema de escalabilidade do CCQSP.

O CCQSP é um problema combinatorial e de difícil solução. Por ser um problema novo, a literatura que trata o CCQSP ainda carece de alternativas eficientes para lidar com o grande número de combinações possíveis de soluções do problema. A fonte do problema reside nas variáveis binárias e inteiras do modelo a ser resolvido. A partir do modelo descrito na seção 5.4.1 e adicionando as melhorias propostas na seção 5.5, haverá

apenas as variáveis binárias  $Y_{at}^{\mathbb{I}k}$  e  $Y_{at}^{\mathbb{O}i}$  no modelo com  $k \in C_j^{\mathbb{I}}$  representando os índices das regiões convexas que modelam os pontos de passagem  $j \in N_a^{\mathbb{I}}$  do episódio  $a$ . Temos  $i \in H_k^{\mathbb{O}}$  com  $k \in C_j^{\mathbb{O}}$  e  $j \in N_a^{\mathbb{O}}$  que são os índices dos hiperplanos  $i$  da região convexa  $k$ , que modelam os obstáculos  $j \in N_a^{\mathbb{O}}$ . Para facilitar a análise da complexidade, os dois conjuntos de variáveis serão reorganizados como proposto abaixo para melhor visualização dos índices envolvidos.

$$Y_{at}^{\mathbb{I}k} \Leftrightarrow Y_{atjk}^{\mathbb{I}} \quad \forall (a, t, j \in N_a^{\mathbb{I}}, k \in C_j^{\mathbb{I}}) \quad (5.74)$$

$$Y_{at}^{\mathbb{O}i} \Leftrightarrow Y_{atjki}^{\mathbb{O}} \quad \forall (a, t, j \in N_a^{\mathbb{O}}, k \in C_j^{\mathbb{O}}, i \in H_k^{\mathbb{O}}) \quad (5.75)$$

Vamos considerar também a cardinalidade de cada conjunto de índices:

$|\mathcal{A}|$  Número total de episódios.

$|T|$  Número total de time steps (waypoints).

$|N|$  Regiões não convexas que definem pontos de passagem e obstáculos.

$|C|$  Regiões convexas

$|H|$  Hiperplanos que definem as regiões convexas.

Considerando apenas as variáveis binárias como base para resolver o problema e calculando todas as possibilidades, teríamos um problema combinatorial de complexidade  $\mathcal{O}[2^{|\mathcal{A}| \cdot |T| \cdot |N| \cdot |C|}] \cdot \mathcal{O}[2^{|\mathcal{A}| \cdot |T| \cdot |N| \cdot |C| \cdot |H|}]$  a partir de  $Y_{atjk}^{\mathbb{I}}$  e  $Y_{atjki}^{\mathbb{O}}$ . Por simplicidade, denotaremos nesta seção os índices apenas por  $(A, T, N, C, H)$  e, para auxiliar na análise, sempre deixaremos separados as complexidades dos dois grupos de variáveis binárias. A primeira complexidade representará a variável envolvida com desvio de obstáculos ( $\mathbb{O}$ ) e multiplica a segunda complexidade atribuída aos pontos de passagem. Apesar da primeira complexidade dominar a segunda, não faremos a simplificação e manteremos elas separadas, uma vez que os índices utilizados em cada uma apresentam valores diferentes.

Por exemplo, vamos considerar um cenário simples como na Figura 4 com 3 obstáculos ( $N^{\mathbb{O}} = 3$ ), 5 pontos de passagem ( $N^{\mathbb{I}} = 5$ ), ambos definidos apenas por regiões convexas únicas ( $C = 1$ ) contendo 4 hiperplanos cada ( $H = 4$ ) e utilizando 10 time steps ( $T = 10$ ). Teremos 6 episódios com 5 modelando os pontos de passagem  $A^{\mathbb{I}} = 5$  e 1 modelando o desvio de obstáculos  $A^{\mathbb{O}} = 1$ . Nesse caso o número de combinações do problema será  $2^{1 \cdot 10 \cdot 3 \cdot 1 \cdot 4} \cdot 2^{5 \cdot 10 \cdot 1 \cdot 1} = 2^{120} \cdot 2^{50} = 2^{170}$ .

Retomando nossa análise, vamos adicionar as restrições já existentes no modelo a fim de reduzir a complexidade exponencial desse problema e amortecer ao máximo

este crescimento. Uma vez que o desvio de obstáculos parece ser o grande gargalo deste problema, atacaremos primeiro a variável de desvio de obstáculos  $Y_{at}^{\circ}$ . Na seção 5.5.3.4 foi apresentado o princípio da exclusão mútua para definir o desvio de obstáculos mais eficientemente, onde a equação 5.72 apresenta o modelo padrão e a equação 5.73 apresenta a exclusão mútua. Considerando isoladamente cada restrição, a complexidade das suas variáveis será  $\mathcal{O}[2^H]$  e  $\mathcal{O}[H]$ , respectivamente. A exclusão mútua faz com que apenas um dos hiperplanos seja selecionado, logo haverá apenas  $H$  possibilidades quando  $Y_{at} = 1$  e nenhuma possibilidade quando  $Y_{at} = 0$ . O pior cenário para a restrição 5.73 é  $\mathcal{O}[H]$ . A restrição 5.72 permite um pior cenário de  $\mathcal{O}[2^H]$ , pois praticamente todas as combinações são válidas quando  $Y_{at} = 1$ .

$$Y_{at} \leq \sum_{i \in H_k^{\circ}} Y_{atjki}^{\circ} \quad \forall (a, t, j \in N_a^{\circ}, k \in C_j^{\circ}) \quad (5.72)$$

$$Y_{at} = \sum_{i \in H_k^{\circ}} Y_{atjki}^{\circ} \quad \forall (a, t, j \in N_a^{\circ}, k \in C_j^{\circ}) \quad (5.73)$$

Sendo assim, com a implementação da exclusão mútua, a complexidade geral dos obstáculos será reduzida de  $\mathcal{O}[2^{A \cdot T \cdot N \cdot C \cdot H}]$  para  $\mathcal{O}[H^{A \cdot T \cdot N \cdot C}]$ . Considerando o exemplo anterior, estaremos reduzindo o número de combinações dos obstáculos de  $2^{120}$  para  $4^{1 \cdot 10 \cdot 3 \cdot 1} = 2^{60}$ .

Os índices  $A$ ,  $T$ ,  $N$  e  $C$  permanecem na parte exponencial. Normalmente, para o desvio de obstáculos, é comum  $A = 1$  ou recebe valores baixos em aplicações práticas. Por outro lado,  $N$  poderá ser alto, já que representa o número de obstáculos. Isso indica que resolver cenários com grande número de obstáculos será um problema. Temos  $C = 1$ , uma vez que os obstáculos, normalmente, são definidos como convexos. Por último, deseja-se que  $T$  seja o mais alto possível para aplicações práticas; caso contrário, teríamos que fazer planejamento de missões de curta duração, o que limita o escopo das aplicações reais. Nesse caso, outra saída seria utilizar uma discretização ruim de  $\Delta t$ , o que prejudica a otimalidade do modelo ao perder a precisão da equação de dinâmica. Isto posto, vamos tratar inicialmente  $T$  com o uso das equações abaixo:

$$S_{atjki} \geq Y_{a,t+1,jki}^{\circ} - Y_{atjki}^{\circ} \quad \forall (a, t, j \in N_a^{\circ}, k \in C_j^{\circ}, i \in H_k^{\circ}) \quad (5.76)$$

$$\sum_t \sum_{i \in H_k^{\circ}} S_{atjki} \leq \bar{S} \quad \forall (a, j \in N_a^{\circ}, k \in C_j^{\circ}) \quad (5.77)$$

Uma variável contínua  $S_{atjki}$  é adicionada ao modelo e a equação 5.76 estabelece quando ocorreram as trocas de ativação entre os hiperplanos. O motivo disso é que, considerando um horizonte de planejamento grande, sabe-se que em cada obstáculo  $j$  o

veículo estará com um único hiperplano ativo durante vários instantes  $t$ . Haverá trocas de hiperplanos apenas quando o veículo inicia o contorno do obstáculo. Em muitas aplicações práticas, é esperado que o veículo não fique ciclando em volta de um obstáculo sem propósito definido. Portanto, podemos limitar as possibilidades de ciclagem estabelecendo um novo parâmetro  $\bar{S}$ , que em conjunto com a equação 5.77 estabelecerá o número máximo de trocas para o cumprimento da missão. Vale lembrar que este número deve ser escolhido empiricamente com base no conhecimento do cenário e seguindo o seguinte princípio: quanto menor melhor. Todavia, se ele for pequeno demais poderá excluir a solução ótima do conjunto de combinações ou tornar o problema infactível. Um valor aceitável em muitas aplicações práticas é definir  $\bar{S} = H$ , pois estaremos permitindo que o veículo contorne cada obstáculo no máximo uma vez.

Analisando a complexidade das equações 5.76, 5.77 em conjunto com 5.73, fica claro que o número máximo de trocas dentro do horizonte de planejamento é  $\bar{S}$ . A matriz  $Y_{atjki}^{\circ}$  deverá manter seu estado quase sem mudanças, mas, serão permitidas dentro do horizonte de planejamento  $t$  e hiperplanos  $i \in H_k^{\circ}$ , apenas  $\bar{S}$  alterações.

Logo, haverá em cada alteração  $\mathcal{O}[H \cdot T]$  possibilidades de escolha e como há um total de  $\bar{S}$  alterações teremos uma complexidade  $\mathcal{O}[(H \cdot T)^{\bar{S}}]$  a cada obstáculo definido. A complexidade geral de todos os obstáculos será agora de  $\mathcal{O}[(H \cdot T)^{A \cdot N \cdot C \cdot \bar{S}}]$ . Nota-se que retiramos o termo  $T$  da parte exponencial, mas adicionamos  $\bar{S}$ . A vantagem dessa troca foi que o parâmetro  $\bar{S}$  pode ser melhor controlado e não depende do horizonte de planejamento da missão, sendo uma constante baseada na complexidade do cenário. Após uma boa escolha de  $\bar{S}$ , pode-se definir livremente qualquer valor de  $T$  que o problema continuará sendo factível e sem exclusão da solução ótima independente do  $T$  escolhido.

Retornando ao nosso exemplo, escolhendo o valor padrão de  $\bar{S} = H = 4$ , o número de combinações dos obstáculos agora será  $(4 \cdot 10)^{1 \cdot 3 \cdot 1 \cdot 4} \approx 2^{63.86}$ . À primeira vista, é um resultado pior que o anterior de  $2^{60}$ , mas agora há maior liberdade para aumentar o valor de  $T$  em crescer muito a complexidade. Vamos supor que desejamos resolver o mesmo problema com  $T = 20$ . Fazendo os cálculos, teremos um resultado melhor, em que a complexidade anterior subirá para  $2^{120}$  e a nova complexidade será  $2^{75.86}$ . Em resumo, o aumento no horizonte de planejamento não afetará mais exponencialmente a complexidade no desvio de obstáculos.

Este princípio pode ser melhorado substituindo a equação 5.77 anterior pela nova equação 5.78. O parâmetro  $\bar{S}$  se tornou um parâmetro global  $\hat{S}$  que é o número total de trocas permitido para todos os obstáculos. Assim, é possível ponderar melhor o valor a ser utilizado, já que é comum uma missão ser cumprida contornando apenas alguns cantos dos obstáculos. A resolução do modelo passa a ficar livre para decidir quais cantos contornar. Novamente, quanto menor for o valor de  $\hat{S}$  melhor. Porém, valores muito baixos podem tornar o problema infactível, em que uma possibilidade é estabelecer  $\hat{S} = 2 \cdot N$ . Tal valor



é razoável, pois indica duas trocas por obstáculo em média, mas o modelo agora é quem ponderará isso uma vez que obstáculos distantes tendem a não ativar nenhuma troca ou no máximo uma única troca. Obstáculos próximos da rota estarão livres para utilizar mais trocas e a média global será menor que 2 trocas por obstáculo. Testes podem ser conduzidos analisando um grande número de missões e rotas para definir adequadamente como o valor de  $\hat{S}$  se relaciona com  $N$ . Assim, pode-se chegar a um parâmetro mais constante e independente de conhecimento humano. Possíveis técnicas que prevejam, para um dado cenário, o valor ideal do parâmetro será uma tarefa de trabalhos futuros. Nesta tese, será escolhido um valor não ideal, mas grande o suficiente para tratar todos os cenários propostos.

$$\sum_a \sum_t \sum_{j \in N_a^\circ} \sum_{k \in C_j^\circ} \sum_{i \in H_k^\circ} S_{atjki} \leq \hat{S} \quad (5.78)$$

Utilizando o mesmo princípio da análise anterior, serão aceitas apenas  $\hat{S}$  mudanças nas variáveis binárias de obstáculo. A mudança poderá acontecer em qualquer posição da matriz. Cada mudança tem  $\mathcal{O}[A \cdot T \cdot N \cdot C \cdot H]$  possibilidades de escolha. Uma vez que, haverá no máximo  $\hat{S}$  mudanças, a complexidade geral para o desvio de obstáculos será de  $\mathcal{O}[(A \cdot T \cdot N \cdot C \cdot H)^{\hat{S}}]$ . Conseguimos, assim, retirar todos os índices da parte exponencial, sendo substituídos por um parâmetro de controle  $\hat{S}$ . Estudos podem ser feitos para verificar como esse parâmetro está relacionado com o número de obstáculos. Se tomarmos  $\hat{S} = 2 \cdot N$  ainda haverá obstáculos na parte exponencial. Retomando o exemplo anterior com  $T = 20$  e  $\hat{S} = 2 \cdot N$  a complexidade agora será de  $(1 \cdot 20 \cdot 3 \cdot 1 \cdot 4)^6 \approx 2^{47.44}$ .

Vamos assumir que existe um valor  $\hat{S}$  constante que resolve o problema, ou que se relaciona com os obstáculos de forma logarítmica. Haverá então uma prova de que o desvio de obstáculo tem complexidade polinomial, mas ainda assim será um polinômio com a multiplicação de 5 parâmetros antes de ser elevado ao valor constante de  $\hat{S}$ . Espera-se que esse polinômio seja de grau elevado e, para reduzir ainda mais a complexidade, há duas possibilidades: tentar reduzir mais o termo exponencial  $\hat{S}$  ou reduzir os termos da base  $(A \cdot T \cdot N \cdot C \cdot H)$ . Em princípio, o termo de maior impacto na base é o parâmetro  $T$  com valores na casa das dezenas, centenas ou até milhares em uma missão de longa duração. A próxima estratégia será, portanto, uma modelagem para remover o  $T$  de dentro do polinômio. Vamos partir de um princípio semelhante ao utilizado na equação 5.78, que definiu um limite para o número de trocas durante o cumprimento de uma missão. Ela também definiu que as trocas não dependem da discretização do horizonte de planejamento, mas sim das manobras no caminho em que a rota ótima está. Redefiniremos as variáveis  $S_{atjki}$  e  $Y_{atjki}^\circ$ , já que o índice  $t$  é o motivo de  $T$  estar na base da complexidade. Não é necessário considerar o tempo no planejamento do desvio de obstáculos e sim as manobras em volta dos obstáculos que é um valor constante. Um novo índice  $s$  será estabelecido,

que representará uma espécie de macro período que indicará a ocorrência das manobras (trocas entre hiperplanos). A partição  $t \in \Phi_s$  é um parâmetro contendo os instantes  $t$  classificados no macro período  $s = 0, \dots, S$  ao tratar o desvio de obstáculos, sendo  $S$  um parâmetro associado a  $\hat{S}$  (número de manobras necessários para completar a missão) e não mais a  $T$  (waypoints).

As variáveis redefinidas passam a ser  $S_{asjki}$  e  $Y_{asjki}^{\circ}$ , onde o índice  $t$  muda para  $s$ . Desta forma, as equações 5.73, 5.76 e 5.78 serão atualizadas para as novas equações:

$$Y_{at} = \sum_{i \in H_k^{\circ}} Y_{asjki}^{\circ} \quad \forall (a, s, t \in \Phi_s, j \in N_a^{\circ}, k \in C_j^{\circ}) \quad (5.79)$$

$$S_{asjki} \geq Y_{a,s+1,jki}^{\circ} - Y_{asjki}^{\circ} \quad \forall (a, s, j \in N_a^{\circ}, k \in C_j^{\circ}, i \in H_k^{\circ}) \quad (5.80)$$

$$\sum_a \sum_s \sum_{j \in N_a^{\circ}} \sum_{k \in C_j^{\circ}} \sum_{i \in H_k^{\circ}} S_{asjki} \leq \hat{S} \quad (5.81)$$

Isso leva a uma nova complexidade uma vez que serão aceitos apenas  $\hat{S}$  mudanças nas variáveis binárias de obstáculo, mas as variáveis binárias estarão agora indexadas apenas em  $s$  e não mais em  $t$ . A matriz  $S_{asjki}$  aceitará no máximo  $\hat{S}$  mudanças, que poderá acontecer em qualquer qualque posição da matriz, onde cada mudança tem  $\mathcal{O}[A \cdot S \cdot N \cdot C \cdot H]$  possibilidades de escolha. A complexidade geral para o desvio de obstáculos será de  $\mathcal{O}[(A \cdot S \cdot N \cdot C \cdot H)^{\hat{S}}]$ . Isso torna possível discretizar o tempo  $T$  sem prejudicar a complexidade no desvio de obstáculos. O parâmetro  $T$  ainda participará na complexidade dos pontos de passagem que será otimizada mais adiante. Uma escolha rigorosa quanto ao número de macro períodos utilizados  $S$  se faz necessária. Se o valor for baixo, não será encontrado o ótimo ou o modelo será infactível. Isso gera um efeito colateral, pois valores baixos de  $S$  limitam os instantes na escolha das manobras e a solução será apenas próxima da ótima (teremos um *upper bound*). Isso ainda será melhor do que a proposta anterior, pois a proposta anterior levava a uma discretização ruim que também tornava a solução final apenas próxima à ótima. As escolhas dos novos parâmetros serão  $\hat{S} \leq S \ll T$ . Retornando ao nosso exemplo, com  $\hat{S} = S = 2 \cdot N$  e agora, independente do valor de  $T$  escolhido, a complexidade será  $(1 \cdot 6 \cdot 3 \cdot 1 \cdot 4)^6 \approx 2^{37.02}$ . A Tabela 8 resume os avanços obtidos com a adição de cada conjunto de equações para melhoria no desvio de obstáculos.

A próxima análise vai se concentrar nas variáveis de pontos de passagem  $Y_{atjk}^{\parallel}$ , que possuem agora maior complexidade que o desvio de obstáculos. Assim, considerando as equações originais descritas na seção 5.4.1 e reescritas abaixo na equação 5.82, podemos perceber que o número de combinações será  $\mathcal{O}[2^{A \cdot T \cdot N \cdot C}]$ . No pior caso,  $Y_{at} = 0$  implica que todas as combinações de  $Y_{atjk}^{\parallel}$  são válidas. No caso de  $Y_{at} = 1$ , também serão todas

Equações	Complexidade	Equivalência na base ( $2^n$ )	Exemplo
5.72	$\mathcal{O}[2^{A \cdot T \cdot N \cdot C \cdot H}]$	$A \cdot T \cdot N \cdot C \cdot H$	$2^{240.00}$
5.73	$\mathcal{O}[(H)^{A \cdot T \cdot N \cdot C}]$	$A \cdot T \cdot N \cdot C \cdot \lg_2  H $	$2^{120.00}$
5.73, 5.76, 5.77	$\mathcal{O}[(H \cdot T)^{A \cdot N \cdot C \cdot \bar{S}}]$	$A \cdot N \cdot C \cdot \bar{S} \cdot \lg_2  H \cdot T $	$2^{75.86}$
5.73, 5.76, 5.78	$\mathcal{O}[(A \cdot T \cdot N \cdot C \cdot H)^{\hat{S}}]$	$\hat{S} \cdot \lg_2  A \cdot T \cdot N \cdot C \cdot H $	$2^{47.44}$
5.79, 5.80, 5.81	$\mathcal{O}[(A \cdot S \cdot N \cdot C \cdot H)^{\hat{S}}]$	$\hat{S} \cdot \lg_2  A \cdot S \cdot N \cdot C \cdot H $	$2^{37.02}$

Tabela 8 – Complexidade para  $\mathbb{O}$  e exemplo com  $T = 20$ ,  $\bar{S} = H$  e  $\hat{S} = S = 2 \cdot N$ 

válidas exceto por um caso, onde tudo é zero. Considerando o caso exemplificado acima para  $T = 20$ , o número de combinações será  $2^{5 \cdot 20 \cdot 1 \cdot 1} = 2^{100}$

$$\left\{ \begin{array}{l} Y_{at} \leq \sum_{j \in N_a^{\mathbb{I}}} Y_{atj}^{\mathbb{I}} \\ Y_{atj}^{\mathbb{I}} \leq \sum_{k \in C_j^{\mathbb{I}}} Y_{atjk}^{\mathbb{I}} \end{array} \quad \forall (j \in N_a^{\mathbb{I}}) \right\} \quad \forall (t, a) \quad (5.82)$$

Na seção 5.5.3.3, o princípio da exclusão mútua dos pontos de passagem foi apresentado, onde a ativação de apenas 1 das regiões convexas  $k \in C_j^{\mathbb{I}}$  de um dos pontos de passagem  $j \in N_a^{\mathbb{I}}$  do episódio  $a$  é condição suficiente para a missão. A equação 5.83 apresenta a exclusão mútua e sua complexidade será  $\mathcal{O}[(N \cdot C)^{A \cdot T}]$ . Se  $Y_{at} = 0$ , então haverá apenas uma combinação válida, que é o caso onde  $Y_{atjk}^{\mathbb{I}} = 0$ . Se  $Y_{at} = 1$ , então haverá para cada  $a, t$  uma complexidade de  $\mathcal{O}[N \cdot C]$ , já que uma e somente uma variável binária está ativa para todo  $j \in N_a^{\mathbb{I}}$  e  $k \in C_j^{\mathbb{I}}$ . Por fim, a complexidade de  $\mathcal{O}[N \cdot C]$  será combinada  $A \cdot T$  vezes resultando na parte exponencial da complexidade informada acima.

$$\left\{ \begin{array}{l} Y_{at} = \sum_{j \in N_a^{\mathbb{I}}} Y_{atj}^{\mathbb{I}} \\ Y_{atj}^{\mathbb{I}} = \sum_{k \in C_j^{\mathbb{I}}} Y_{atjk}^{\mathbb{I}} \end{array} \quad \forall (j \in N_a^{\mathbb{I}}) \right\} \quad \forall (t, a) \quad (5.83)$$

Deve-se ressaltar um caso especial que vale para todas as análises informadas anteriormente. Se a multiplicação dos termos da base resultar no número 1, então se considera base como sendo 2. A análise da complexidade se preocupa com o crescimento de maior grau apenas, pois ele que dominará a complexidade. Todavia, para um exemplo pequeno como o que estamos utilizando e onde alguns parâmetros tem valor de apenas uma unidade, a exceção mencionada pode ocorrer. No nosso exemplo, para  $T = 20$ , o número de combinações agora seria  $(1 \cdot 1)^{5 \cdot 20} = 1$  e devemos considerar a base mínima 2, visto que havia duas condições ( $Y_{at} = 0$  e  $Y_{at} = 1$ ) e logo o número mínimo de combinações da base é 2. Portanto, teremos  $2^{5 \cdot 20} = 2^{100}$  que é de fato o valor da exclusão mútua quando não há nada a excluir, como ocorreu no exemplo com  $N$  e  $C$  valendo 1. Por isso, a complexidade do exemplo permaneceu a mesma. O motivo apresentado para a exceção é coerente porque, neste caso, a matriz  $Y_{atjk}^{\mathbb{I}}$  terá uma única célula nos índices  $[j, k]$ . Haverá duas possibilidades de valores (0 ou 1) levando a uma base 2 com as duas possibilidades sendo combinadas  $A \cdot T$  vezes. Ainda assim, a exclusão mútua deve ser implementada

para garantir uma complexidade menor em casos mais gerais que não estão nesta situação particular.

Vamos analisar agora o impacto das melhorias propostas na seção 5.5.3.1, onde é estabelecido que para os episódios do tipo Start-In e End-In vale a equação 5.66 e para episódios do tipo Remain-In vale a equação 5.67

$$\sum_t Y_{at} = 1 \quad \forall (a \in \mathcal{A}^{S|E}) \quad (5.66)$$

$$\left\lceil \frac{\mathcal{T}_r^{lb}}{\Delta t} \right\rceil \leq \sum_t Y_{at} \leq \left\lfloor \frac{\mathcal{T}_r^{ub}}{\Delta t} \right\rfloor \quad \forall (a \in \mathcal{A}^R, r \in \mathcal{T} | e_a^S = e_r^S \wedge e_a^E = e_r^E) \quad (5.67)$$

Há duas complexidades possíveis. A primeira será para episódios Start-In e End-In, onde podemos perceber que a adição da equação 5.66 estabelece a seleção de um e apenas um instante  $t$  no qual o episódio estará ativo. O impacto na complexidade será de retirar o  $T$  da exponencial e teremos uma complexidade final de  $\mathcal{O}[(T \cdot N \cdot C)^A]$  para  $a \in \mathcal{A}^{S|E}$ . A prova disso é simples de perceber, pois com a seleção de um único instante pela equação 5.66 teremos também uma única região convexa selecionada para este instante pela equação 5.83. Assim, para cada  $a \in \mathcal{A}^{S|E}$ , a complexidade de selecionar uma única variável binária para ativação dentro de um subconjunto da matriz  $Y_{atjk}^{\mathbb{I}}$  nos índices  $t, j, k$  será  $\mathcal{O}[T \cdot N \cdot C]$ . Combinado  $A$  vezes, considerando os episódios  $S|E$ , resultará na complexidade  $\mathcal{O}[(T \cdot N \cdot C)^A]$  para  $a \in \mathcal{A}^{S|E}$ . No exemplo considerado, temos  $(20 \cdot 1 \cdot 1)^5 \approx 2^{21.61}$ .

A segunda complexidade considera episódios do tipo Remain-In para os quais a equação 5.67 está estabelecida. Nesse caso, temos uma complexidade  $\mathcal{O}[(T \cdot N \cdot C)^{A \cdot \mathcal{T}}]$ , onde o parâmetro  $\mathcal{T}$  representa simplificadaamente a duração máxima do episódio  $\mathcal{T} = \left\lfloor \frac{\mathcal{T}_r^{ub}}{\Delta t} \right\rfloor$ . Teremos em geral  $\mathcal{T} \ll T$  para aplicações práticas, uma vez que a duração máxima da execução de uma tarefa individual costuma ser bem inferior ao tempo total da missão. Em nosso exemplo, não há episódios do tipo Remain-In para pontos de passagem. Contudo, para ilustrar seu impacto na complexidade, vamos supor a adição de um único episódio do tipo Remain-In entre os 5 episódios End-In do exemplo. Esse episódio Remain-In terá duração de  $\mathcal{T} = 6$ . Assim, a complexidade seria  $(20 \cdot 1 \cdot 1)^{1 \cdot 6} \approx 2^{25.93}$  que é um valor maior do que a complexidade conjunta de todos os outros 5 episódios. Logo, episódios do tipo Remain-In possuem um impacto negativo na complexidade. Diante disso, buscaremos, a seguir, novas alternativas.

A proposta será substituir a equação 5.67 pelas equações 5.84 até 5.88, em que duas novas variáveis contínuas são adicionadas:  $S_{at}^{\mathbb{I}\uparrow}$  e  $S_{at}^{\mathbb{I}\downarrow}$ . Essas variáveis, apesar de serem contínuas, sempre receberão valores 0 ou 1 já que são estabelecidas a partir da variável  $Y_{at}$

com valor 0 ou 1.

$$S_{at}^{\mathbb{I}\uparrow} \geq Y_{at}^{\mathbb{I}} - Y_{a,t-1}^{\mathbb{I}} \quad \forall(a, t) \quad (5.84)$$

$$S_{a,t+1}^{\mathbb{I}\uparrow} \geq S_{at}^{\mathbb{I}\uparrow} \quad \forall(a, t) \quad (5.85)$$

$$S_{at}^{\mathbb{I}\downarrow} \geq Y_{a,t-1}^{\mathbb{I}} - Y_{at}^{\mathbb{I}} \quad \forall(a, t) \quad (5.86)$$

$$S_{a,t+1}^{\mathbb{I}\downarrow} \geq S_{at}^{\mathbb{I}\downarrow} \quad \forall(a, t) \quad (5.87)$$

$$Y_{at} = S_{at}^{\mathbb{I}\uparrow} \wedge \neg S_{at}^{\mathbb{I}\downarrow} \Leftrightarrow \left\{ \begin{array}{l} Y_{at} \leq S_{at}^{\mathbb{I}\uparrow} \\ Y_{at} \leq 1 - S_{at}^{\mathbb{I}\downarrow} \\ Y_{at} \geq S_{at}^{\mathbb{I}\uparrow} - S_{at}^{\mathbb{I}\downarrow} \end{array} \right\} \quad \forall(t, a) \quad (5.88)$$

As equações 5.84 e 5.86 marcam, respectivamente, a subida ( $\uparrow$ ) e descida ( $\downarrow$ ) do sinal de  $Y_{at}$  com o fluxo do tempo  $t$ . As equações 5.85 e 5.87 estabelecem que, após capturada a subida ou decida, o sinal deve permanecer, respectivamente, em estado alto ou baixo até o fim do horizonte de planejamento. Finalmente, a equação 5.88 estabelece o fechamento garantindo que: 1) exista apenas um instante no qual o sinal de  $Y_{at}$  sobe; 2) exista apenas um instante no qual o sinal de  $Y_{at}$  desce; 3) entre todos os instantes da subida até a descida,  $Y_{at}$  permanecerá obrigatoriamente ativado e estará desativado em todos os outros instantes em que isto não ocorre. Com essa lógica implementada, temos um fechamento bem forte imposto pelas restrições que permitem apenas combinações válida.

Ao analisar a complexidade da nova proposta perceberemos que uma vez  $Y_{at}$  ativado para um episódio Remain-In, esse permanecerá ativo até terminar o episódio. Logo, teremos  $Y_{at} = 0$  antes da ocorrência do episódio, depois  $Y_{at} = 1$  durante o episódio e  $Y_{at} = 0$  novamente para todos os instantes depois do episódio. Como os instantes em que ocorrem a ativação e a desativação são tomadas de decisão livres, haverá uma complexidade em princípio de  $\mathcal{O}[T \cdot T]$  para essas escolhas, pois existem  $T$  escolhas para iniciar a ativação e  $T$  escolhas para terminar a ativação. A primeira escolha reduz o escopo da segunda escolha e podemos melhorar a dedução da complexidade com base nas durações. Se considerarmos a existência da equação 5.12 (descrita na seção 5.4.1 e repetida abaixo) e tomarmos um novo parâmetro  $\widehat{\mathcal{T}} = \left\lfloor \frac{\mathcal{T}_r^{ub}}{\Delta t} \right\rfloor - \left\lceil \frac{\mathcal{T}_r^{lb}}{\Delta t} \right\rceil$ , então teremos  $T$  possibilidade para a primeira escolha com apenas  $\widehat{\mathcal{T}}$  possibilidades para escolher a descida do sinal. Isso faz sentido, pois uma vez escolhido o ponto de início, o ponto de final estará entre os valores da duração mínima ( $\mathcal{T}_r^{lb}$ ) e máxima ( $\mathcal{T}_r^{ub}$ ) do episódio.

A complexidade será  $\mathcal{O}[T \cdot \widehat{\mathcal{T}}]$  para as escolhas do sinal em cada episódio do tipo Remain-In e a complexidade final de todos os episódios Remain-In será  $\mathcal{O}[(T \cdot \widehat{\mathcal{T}} \cdot N \cdot C)^A]$ , considerando que a ativação de  $Y_{at}$  implica também em escolha dos pontos de passagem convexos. Isso se justifica, uma vez que ao considerarmos um episódio Remain-In com duração de uma unidade de tempo apenas (ou seja  $\widehat{\mathcal{T}} = 1$ ), a complexidade será igual a dos episódios Start-In e End-In. No caso do exemplo com um episódio Remain-In considerando  $\widehat{\mathcal{T}} = 6$ , a complexidade isolada passa de  $2^{25.93}$  para  $(20 \cdot 6 \cdot 1 \cdot 1)^1 \approx 2^{6.91}$ .

$$\mathcal{T}_r^{lb} \leq T_{e_r^E} - T_{e_r^S} \leq \mathcal{T}_r^{ub} \quad \forall (r \in \mathcal{T}) \quad (5.12)$$

Em resumo, para os pontos de passagem, teremos uma complexidade combinada de  $\mathcal{O}_{S|E}[(T \cdot N \cdot C)^{A_{S|E}}] \cdot \mathcal{O}_R[(T \cdot \widehat{\mathcal{T}} \cdot N \cdot C)^{A_R}]$ , em que  $\mathcal{O}_{S|E}[(T \cdot N \cdot C)^{A_{S|E}}]$  é a complexidade para episódios Start-In e End-In e  $\mathcal{O}_R[(T \cdot \widehat{\mathcal{T}} \cdot N \cdot C)^{A_R}]$  para episódios Remain-In. Notemos que há apenas um termo na parte exponencial que se refere ao tipo de episódio ( $A_{S|E}$  ou  $A_R$ ). Logo, podemos reescrever a complexidade como  $\mathcal{O}[(T \cdot N \cdot C)^A \cdot (\widehat{\mathcal{T}}^{A_R})]$  com  $A$  indicando o número total de episódios, considerando todos os tipos,  $A_R$  é o número de episódios do tipo Remain-In e  $\widehat{\mathcal{T}}$  representa quantos instantes existem na diferença entre a duração máxima e mínima do episódio  $A_R$  para uma discretização  $\Delta t$ . O problema passa a ser exponencial apenas com o acréscimo dos episódios que definem os pontos de passagem, onde uma complexidade extra ocorre se o episódio for do tipo Remain-In. A complexidade será pior quanto maior for a diferença no tempo de permanência nessa região. O número de episódios pode chegar a casa das dezenas para missões de complexidade média. Logo, para uma missão com um número de episódios fixos, a complexidade resultante será um polinômio de grau elevado.

Um fator que possui impacto relevante está novamente relacionado ao parâmetro  $T$ , que domina os valores na base do polinômio, caso se considere um horizonte de planejamento o maior possível. Uma redefinição das variáveis  $Y_{atjk}^I$  e  $Y_{at}$  para  $Y_{asjk}^I$  e  $Y_{as}$  será considerada, em que também utilizaremos o mesmo conceito de macro período dos obstáculos, que agora passa a ser macro período dos pontos de passagem. Logo,  $s = 0..S$  e  $t \in \Phi_s$  são os parâmetros contendo os instantes  $t$  classificados no macro período  $s$ . Os macro períodos referentes aos pontos de passagem podem ser separados dos macro períodos referentes ao desvio de obstáculos, porém isto tornaria mais difícil de desenvolver a modelagem. Numa primeira versão, será considerado que não há necessidade desta separação. Assim, poderemos contar com a simplicidade dos períodos  $t$  estarem sincronizados dentro de cada macro período  $s$  tanto para pontos de passagem quanto para obstáculos. Com a redefinição destas novas variáveis, teremos que estabelecer novamente algumas equações. As equações originais 5.11, 5.15, 5.16, 5.17, 5.18 passam a ser, respectivamente, as equações 5.89, 5.90,

5.91, 5.92, 5.93.

$$T_{e+1} = T_e + k_e \cdot \Delta t' \quad \forall(e) \quad (5.89)$$

$$T_{e_a^S} = s \cdot \Delta t' \Rightarrow Y_{as} \quad \forall(s, a \in \mathcal{A}^S) \quad (5.90)$$

$$T_{e_a^S} \leq s \cdot \Delta t' \leq T_{e_a^E} \Rightarrow Y_{as} \quad \forall(s, a \in \mathcal{A}^R) \quad (5.91)$$

$$T_{e_a^E} = s \cdot \Delta t' \Rightarrow Y_{as} \quad \forall(s, a \in \mathcal{A}^E) \quad (5.92)$$

$$Pr \left[ \bigwedge_{a \in \Psi_c} \bigwedge_s \left( Y_{as} \Rightarrow \left\{ \begin{array}{l} \mathbf{x}_t \in R_a \quad a \in \mathcal{A}^S \wedge t = first(\Phi_s) \\ \mathbf{x}_t \in R_a \quad a \in \mathcal{A}^R \wedge \forall(t \in \Phi_s) \\ \mathbf{x}_t \in R_a \quad a \in \mathcal{A}^E \wedge t = last(\Phi_s) \end{array} \right\} \right) \right] \geq 1 - \Delta_c \quad \forall(c) \quad (5.93)$$

O parâmetro de intervalo de tempo  $\Delta t' = |\Phi_s| \cdot \Delta t$  representa a discretização dos macro períodos de tempo que serão considerados na ativação dos pontos de passagem e manobras para os obstáculos. O parâmetro  $\Delta t$  permanecerá sendo utilizado apenas na equação de dinâmica para definir uma trajetória mais contínua possível para o veículo. As equações 5.83, 5.66, 5.84 - 5.88 serão atualizadas trocando o índice  $t$  por  $s$  e o parâmetro  $\widehat{T}' = \lfloor \frac{T_r^{ub}}{\Delta t'} \rfloor - \lfloor \frac{T_r^{lb}}{\Delta t'} \rfloor$  agora passa a ser definido pela discretização  $\Delta t'$ . Isso leva ao valor  $\widehat{T}' \ll T$ . A complexidade será semelhante a anterior, mas o parâmetro  $T$  é agora substituído por  $S$ . Assim, a complexidade será:  $\mathcal{O}[(S \cdot N \cdot C)^A \cdot (\widehat{T}'^{A_R})]$ .

Em nosso exemplo, adicionado do episódio Remain-In e considerando  $\widehat{S} = S = 2 \cdot N$ , teremos  $\widehat{T}' = 2$ , que resultará em uma complexidade  $(6 \cdot 1 \cdot 1)^6 \cdot (2)^1 \approx 2^{16.51}$ . A complexidade agora é independente da discretização em  $T$  para os pontos de passagem. A discretização do tempo não mais aumenta a complexidade das variáveis de obstáculos e pontos de passagem.

A Tabela 9 resume os avanços obtidos na complexidade dos pontos de passagem para cada conjunto de equações.

A complexidade final do problema será agora estabelecida. Para cada combinação considerada, há também a complexidade de resolver um problema de programação linear (PL). O PL pode ser resolvido eficientemente com poucas iterações em  $\mathcal{O}[\sqrt{n} \cdot \ln|1/\epsilon|]$  por um método de pontos interiores (GONDZIO, 2012), em que  $\epsilon$  é um parâmetro que define a acurácia da solução. Para cada iteração do método de pontos interiores, é preciso resolver um sistema matricial, o que leva a complexidade total do método a ser polinomial



Equações	Complexidade	Equivalência na base ( $2^n$ )	Exemplo
5.82	$\mathcal{O}[2^{A \cdot T \cdot N \cdot C}]$	$A \cdot T \cdot N \cdot C$	$2^{120.00}$
5.83	$\mathcal{O}[(N \cdot C)^{A \cdot T}]$	$A \cdot T \cdot \lg_2 N \cdot C $	$2^{120.00}$
5.83, 5.66, 5.67	$\mathcal{O}[(T \cdot N \cdot C)^{A_{S E} + A_R \cdot T}]$	$(A_{S E} + A_R \cdot T) \cdot \lg_2 T \cdot N \cdot C $	$2^{47.54}$
5.83, 5.66, 5.84 - 5.88	$\mathcal{O}[(T \cdot N \cdot C)^A \cdot (\widehat{\mathcal{T}}^{A_R})]$	$A \cdot \lg_2 T \cdot N \cdot C  + A_R \cdot \lg_2 \widehat{\mathcal{T}} $	$2^{28.51}$
5.83, 5.66, 5.84 - 5.93	$\mathcal{O}[(S \cdot N \cdot C)^A \cdot (\widehat{\mathcal{T}}^{A_R})]$	$A \cdot \lg_2 S \cdot N \cdot C  + A_R \cdot \lg_2 \widehat{\mathcal{T}} $	$2^{16.51}$

Tabela 9 – Complexidade para  $\mathbb{I}$  e exemplo com  $T = 20$ ,  $A = 6$ ,  $A_R = 1$ ,  $\mathcal{T} = \widehat{\mathcal{T}} = 6$ ,  $\widehat{S} = S = 2 \cdot N$ ,  $\widehat{\mathcal{T}}' = 2$

em  $\mathcal{O}[n^{3.5} \cdot \ln|1/\epsilon|]$ . O parâmetro  $n$  é o número de variáveis e restrições no modelo. No nosso problema, este valor era dominado por  $\mathcal{O}[A^{\mathbb{O}} \cdot T \cdot N^{\mathbb{O}} \cdot C^{\mathbb{O}} \cdot H + A^{\mathbb{I}} \cdot T \cdot N^{\mathbb{I}} \cdot C^{\mathbb{I}}]$  e após as melhorias apresentadas passou a ser  $\mathcal{O}[T + A^{\mathbb{O}} \cdot S \cdot N^{\mathbb{O}} \cdot C^{\mathbb{O}} \cdot H + A^{\mathbb{I}} \cdot S \cdot N^{\mathbb{I}} \cdot C^{\mathbb{I}}]$ .

Assim, a complexidade final combinada de todo o problema será a complexidade dos obstáculos multiplicada pela complexidade dos pontos de passagem e multiplicada pela complexidade do LP. Considerando os melhores resultados obtidos anteriormente temos:  $\mathcal{O}[(A^{\mathbb{O}} \cdot S \cdot N^{\mathbb{O}} \cdot C^{\mathbb{O}} \cdot H)^{\widehat{S}}] \cdot \mathcal{O}[(S \cdot N^{\mathbb{I}} \cdot C^{\mathbb{I}})^{A^{\mathbb{I}}} \cdot (\widehat{\mathcal{T}}'^{A_R^{\mathbb{I}}})] \cdot \mathcal{O}[(T + A^{\mathbb{O}} \cdot S \cdot N^{\mathbb{O}} \cdot C^{\mathbb{O}} \cdot H + A^{\mathbb{I}} \cdot S \cdot N^{\mathbb{I}} \cdot C^{\mathbb{I}})^{3.5} \cdot \ln|1/\epsilon|]$ . No exemplo apresentado teremos:  $(1 \cdot 6 \cdot 3 \cdot 1 \cdot 4)^6 \cdot (6 \cdot 1 \cdot 1)^6 \cdot (2^1) \cdot (20 + 1 \cdot 6 \cdot 3 \cdot 1 \cdot 4 + 6 \cdot 6 \cdot 1 \cdot 1)^{3.5} \cdot \ln|1/10^{-6}| \approx 2^{37.02} \cdot 2^{16.51} \cdot 2^{24.5} \cdot 2^{19.93} \approx 2^{98}$ . O valor  $2^{98}$  não deve ser considerado em termos absolutos, ele é apenas uma referência para entendermos de onde partimos em  $2^{240} \cdot 2^{120} \cdot 2^{29.75} \cdot 2^{19.93} = 2^{409.68}$  para chegarmos em  $2^{98}$  com as melhorias apresentadas.

O resultado teórico principal é definir um limite superior de como ocorre o crescimento na complexidade do problema. Assim, vamos tomar  $K$  como uma constante quando fixamos todos os parâmetros, exceto por um deles. Em seguida, analisamos o comportamento do crescimento para cada parâmetro individualmente. O resultado mais impressionante é que o horizonte de planejamento  $T$  afeta a complexidade num pior caso por um polinômio de baixo grau  $\mathcal{O}[T^{3.5}]$ . Logo, um cenário considerando um longo horizonte de planejamento pode gerar um problema tratável. Na prática, o resultado esperado deve ser ainda melhor, visto que esse é o pior caso. Essa complexidade reflete apenas a dificuldade dos métodos que resolvem o LP e no geral LPs são rápidos de serem resolvidos.

O número de manobras  $\widehat{S}$  afetará a complexidade exponencialmente  $\mathcal{O}[K^{\widehat{S}}]$ , sendo um parâmetro relativo à complexidade do cenário. Assim, cenários que colocam o veículo em ambientes parecidos com labirintos, onde o único caminho possível leva o veículo a fazer muitas manobras para atravessar, são um problema. Todavia, em muitas situações práticas, os cenários são mais abertos e possuem vários caminhos que levam ao mesmo destino. Isso possibilita que o problema seja resolvido com um número menor de manobras (valor baixo de  $\widehat{S}$ ).

O número de episódios para os pontos de passagem  $A^{\mathbb{I}}$  também afetará a complexi-



dade exponencialmente  $\mathcal{O}[K^{\hat{A}}]$ . Esse parâmetro é relativo a complexidade da missão, visto que missões envolvendo muitas regiões de passagem podem ser um problema. Os demais parâmetros como, número de obstáculos ( $N, C$ ) e hiperplanos ( $H$ ), afetam a complexidade de forma polinomial:  $\mathcal{O}[N^K]$ ,  $\mathcal{O}[C^K]$  e  $\mathcal{O}[H^K]$ . Nesse caso, o polinômio pode ter um grau bem elevado.

Resultados preliminares foram obtidos e mostraram uma drástica redução no tempo computacional para resolver o problema. Testes futuros serão realizados para verificar se as deduções das complexidades apresentadas se refletem nos resultados práticos.



---

# HYBRID CHANCE-CONSTRAINED QUALITATIVE STATE PLAN PROBLEM

---

---

## 6.1 Introdução

Nesta seção, uma proposta de modelo completo para o *Hybrid Chance-Constrained Qualitative State Planning problem* (HCCQSP) é apresentada. O HCCQSP é uma extensão do *Hybrid Qualitative State Planning problem* (HQSP) definido em (LI, 2010). As principais inovações incorporadas ao HQSP por este projeto de doutorado são:

1. Tratamento da incerteza, conforme especificado para o *Chance-Constrained Qualitative State Plan problem* (CCQSP), proposto em (ONO; WILLIAMS; BLACKMORE, 2013).
2. Correção de problemas relacionados a violação de risco nos cantos dos obstáculos, aspecto não tratado em (BLACKMORE; ONO; WILLIAMS, 2011).

Os conceitos utilizados e os tipos de missões tratadas pelo HCCQSP são semelhantes aos estabelecidos no HQSP por (LI, 2010). A Figura 22 exemplifica uma missão tratada no HCCQSP juntamente com uma possível solução. A missão representa um combate a incêndios, sendo informalmente descrita a seguir:

*Iniciando em **base** com o tanque de combustível cheio e reservatório de água vazio, o VANT deve apagar dois focos de incêndio (**fire1** e **fire2**) que estão sobre as regiões de floresta (**forest1** e **forest2**) e, em seguida, retornar para **base**. Não é especificada a ordem de qual incêndio apagar primeiro, mas ambos os focos de incêndio devem ser apagados dentro de um tempo de 20 a 50 minutos e não há restrições de tempo para o retorno à **base**. Toda a missão*

deve ser cumprida em no máximo 60 minutos, respeitando a probabilidade de falha no cumprimento das etapas da missão que é limitada em 1%. O VANT deve permanecer na região segura durante todo o tempo da missão, evitando colisão com os obstáculos NFZ (no-fly zones) dentro de uma margem de risco de 0.0001%. Por precaução, durante a primeira etapa da missão, o combustível deve permanecer acima dos 20% e durante a segunda etapa acima dos 5%

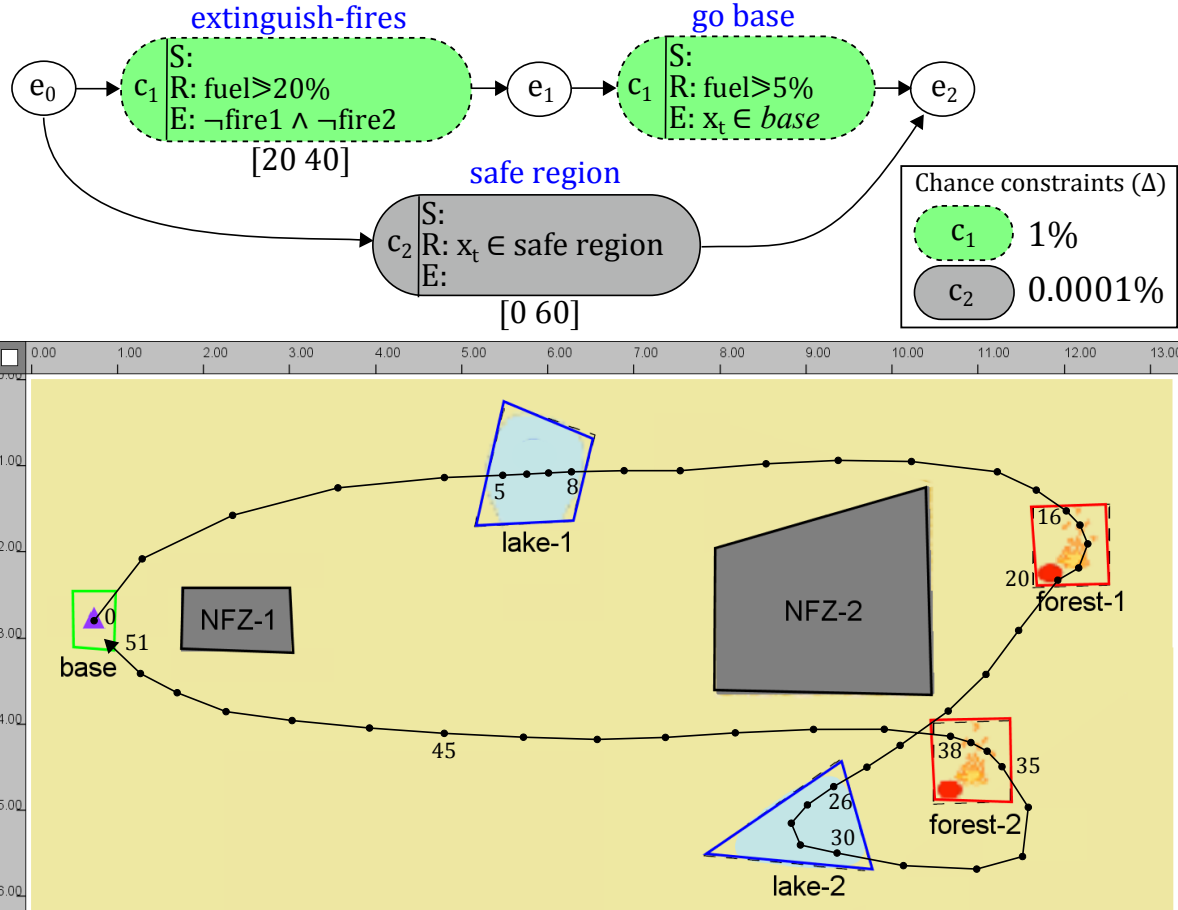


Figura 22 – Missão para o HCCQSP e sua solução considerando  $\Delta t = 1$  minuto, adaptado (LI, 2010)

## 6.2 Descrição do problema

O HCCQSP é formalmente definido a seguir (Def. 6).

**Definição 6** (HCCQSP problem). O HCCQSP consiste em encontrar uma solução híbrida ótima  $\mathcal{S}^*$ , onde:

- $\mathcal{M} = \langle \mathbf{r}, \mathbf{w} \rangle$  é o mapa conhecido do cenário, onde a missão será planejada, contendo regiões mapeadas e sua classificação (Def. 7).
- $\mathcal{X} = \langle \mathbf{x}, \mathbf{u}, \mathbf{l} \rangle$  é um conjunto com variáveis de estado, de controle e literais (Def. 8).

- $\mathcal{I} = \langle \hat{\mathbf{x}}_0, \hat{\mathbf{l}}_0 \rangle$  são as condições iniciais da missão (Def. 9).
- $\mathcal{H}$  são ações híbridas com duração, representando todas as ações que o VANT poderá executar para o cumprimento da missão. (Def. 10).
- $\mathcal{P}$  é o *Chance-Constrained Qualitative State Plan problem* da missão, especificando as etapas a serem cumpridas, sua ordem e os limites de tempo dados a cada etapa bem como as restrições de risco envolvidas. (Def. 15).
- $\mathcal{J}$  é uma função objetivo a ser otimizada na missão.
- $\mathcal{S}^* = \langle \mathcal{H}^*, \mathcal{P}^*, \mathcal{X}^* \rangle$  é a solução híbrida ótima que contém: uma sequencia ótima de ações  $\mathcal{H}^*$  a serem executadas, o *scheduling* ótimo do tempo em que ocorre os eventos em  $\mathcal{P}^*$  e os valores ótimos das variáveis  $\mathcal{X}^*$ .

**Definição 7** (Mapa do cenário).  $\mathcal{M} = \langle \mathbf{r}, \mathbf{w} \rangle$ , onde:

- $\mathbf{r} = \langle r^1, r^2, \dots \rangle$  é o conjunto de regiões  $r \in \mathbf{r}$  mapeadas e de interesse para o cumprimento da missão.
- $\mathbf{w} = \langle w^1, w^2, \dots \rangle$  é um conjunto com todos os tipos  $w \in \mathbf{w}$  no qual uma região  $r \in \mathbf{r}$  está classificada.

Por exemplo, há 7 regiões previamente mapeadas na missão de combate a incêndios,  $\mathbf{r} = \langle NFZ1, NFZ2, base, lake1, lake2, forest1, forest2 \rangle$ . Cada uma delas é classificada em um dos 4 tipos, a seguir  $\mathbf{w} = \langle NFZ, base, lake, forest \rangle$ . Nota-se que toda região  $r \in \mathbf{r}$  é classificada em um e somente um dos tipos  $w \in \mathbf{w}$ , assim denota-se  $r \in \mathbf{r}_w$  como o conjunto de todas as regiões  $r$  classificadas como do tipo  $w$ .

**Definição 8** (Variáveis).  $\mathcal{X} = \langle \mathbf{x}, \mathbf{u}, \mathbf{l} \rangle$ , onde:

- $\mathbf{x} = \langle x^1, x^2, \dots \rangle$  é o conjunto de variáveis reais  $x \in \mathbf{x}$  representando o estado do veículo ( $\mathbf{x}_t$ ) a cada instante de tempo  $t$ .
- $\mathbf{u} = \langle u^1, u^2, \dots \rangle$  é o conjunto de variáveis reais  $u \in \mathbf{u}$  representando o controle do veículo ( $\mathbf{u}_t$ ) a cada instante de tempo  $t$ .
- $\mathbf{l} = \langle l^1, l^2, \dots \rangle$  é o conjunto de variáveis literais  $l \in \mathbf{l}$  representando a situação da missão ( $\mathbf{l}_t$ ) a cada instante de tempo  $t$ .

Por exemplo, na missão de combate a incêndios, as variáveis de estado são posições, velocidades e o nível do combustível  $\mathbf{x} = \langle p^x, p^y, v^x, v^y, fuel \rangle$ . As variáveis de controle são acelerações e a taxa de consumo do combustível  $\mathbf{u} = \langle a^x, a^y, r_f \rangle$ . As variáveis literais são  $\mathbf{l} = \langle fire1, fire2, water \rangle$ .

Tanto as variáveis de estado contínuas  $x \in \mathbf{x}$  quanto as variáveis literais  $l \in \mathbf{l}$  podem ser parametrizadas pelos tipos de região  $w \in \mathbf{w}$ . Logo, a variável literal `fire` é parametrizada pelo tipo `forest` indicando que existem dois focos de incêndio `fire1` e `fire2` nas regiões `forest1` e `forest2`, respectivamente. Denota-se  $l \in \mathbf{l}_w$  o conjunto de todos os literais  $l$  parametrizados pelo tipo  $w$  e  $l \in \mathbf{l}_\emptyset$  o conjunto dos literais não parametrizados. Assim temos,  $\mathbf{l}_{\text{forest}} = \langle \text{fire1}, \text{fire2} \rangle$  e  $\mathbf{l}_\emptyset = \langle \text{water} \rangle$

**Definição 9** (Condições Iniciais).  $\mathcal{I} = \langle \hat{\mathbf{x}}_0, \hat{\mathbf{l}}_0 \rangle$ , onde:

- $\hat{\mathbf{x}}_0$  é o estado do veículo no início do planejamento (instante  $t = 0$ ).
- $\hat{\mathbf{l}}_0$  é o estado dos literais no início do planejamento (instante  $t = 0$ ).

Por exemplo, para a missão de combate a incêndios, as condições iniciais são  $\hat{\mathbf{x}}_0 = \langle 0.7, 2.8, 0, 0, 100\% \rangle$  e  $\hat{\mathbf{l}}_0 = \langle \text{true}, \text{true}, \text{false} \rangle$ . Nesse caso,  $\hat{\mathbf{x}}_0$  indica que o VANT se encontra na `base`, parado e com o tanque de combustível cheio, enquanto  $\hat{\mathbf{l}}_0$  indica que os dois focos de incêndio `fire1` e `fire2` estão ativos (`true`) e o reservatório de água do VANT está vazio (`water = false` ou  $\neg \text{water}$ ).

**Definição 10** (Ações Durativas Híbridas).  $\mathcal{H} = \langle \mathbf{Cond}, \mathbf{Eff}, \mathbf{Dyn}, \mathbf{d} \rangle$ , onde:

- **Cond** é um conjunto de condições necessárias para a ação ser iniciada, mantida e terminada (Def. 11).
- **Eff** são os efeitos discretos causados pela ação (Def. 12).
- **Dyn** são equações de transição de estado (ou equações de dinâmica) e restrições aplicadas durante a execução da ação (Def. 13).
- **d** são os limites de tempo indicando a duração da ação (Def. 14).

A Tabela 10 lista ações durativas híbridas para a missão de combate a incêndios ilustrada na Figura 22. Essas ações serão utilizadas a seguir para facilitar o entendimento

Tabela 10 – Ações Durativas Híbridas para a missão de combate a incêndios

Ação	<code>fly</code>	<code>fill(lake)</code>	<code>drop(forest)</code>
<b>Parâmetros</b>		$r \in \mathbf{r}_{\text{lake}}$	$r \in \mathbf{r}_{\text{forest}}, l \in \mathbf{l}_{\text{forest}}$
<b>Cond</b>	<b>S</b> : <b>R</b> : $\text{fuel} \geq 1\%$ <b>E</b> :	<b>S</b> : $\neg \text{water}$ <b>R</b> : $\mathbf{x}_t \in r$ <b>E</b> :	<b>S</b> : $\text{water}, l$ <b>R</b> : $\mathbf{x}_t \in r$ <b>E</b> :
<b>Eff</b>	<b>S</b> : <b>E</b> :	<b>S</b> : <b>E</b> : $\text{water}$	<b>S</b> : <b>E</b> : $\neg \text{water}, \neg l$
<b>Dyn</b>	eq. 6.3 e 6.4		
<b>d</b>	$[0, +\infty]$	$[4, 8]$	$[2, 5]$

das definições que se seguem.

**Definição 11** (Condições de uma Ação Durativa Híbrida). **Cond** é um conjunto de condições necessárias para a ação ser iniciada, mantida e terminada. Ela pode ser de 3 tipos distintos:

- **Cond<sup>S</sup>** são condições necessárias para a ação ser iniciada (**start**).
- **Cond<sup>R</sup>** são condições necessárias para a ação ser mantida (**remain**).
- **Cond<sup>E</sup>** são condições necessárias para a ação ser terminada (**end**).

Por exemplo, na missão de combate a incêndios, a ação *fly* exige apenas que o VANT tenha o mínimo de combustível para manter o voo, **Cond<sup>R</sup>** =  $\{fuel \geq 1\%\}$ . A ação *fill(lake)* exige que o VANT esteja sem água para iniciar, **Cond<sup>S</sup>** =  $\{\neg water\}$ , e exige que o VANT se mantenha sobre um dos lagos  $r \in \mathbf{r}_{lake}$  durante a coleta de água, **Cond<sup>R</sup>** =  $\{\mathbf{x}_t \in r\}$ . Ao final, a ação *drop(forest)* exige que o VANT tenha água e que um dos focos de incêndio  $l \in \mathbf{l}_{forest}$  esteja ativo para ser iniciada, **Cond<sup>S</sup>** =  $\{water, l\}$ ; exigindo também que durante a ação o VANT se mantenha sobre a região  $r \in \mathbf{r}_{forest}$  onde está o foco de incêndio, **Cond<sup>R</sup>** =  $\{\mathbf{x}_t \in r\}$ .

**Definição 12** (Efeitos discretos de uma Ação Durativa Híbrida). **Eff** é um conjunto de efeitos causados pela execução da ação. Podendo ser de 2 tipos distintos:

- **Eff<sup>S</sup>** se torna verdade no instante em que a ação inicia (**start**).
- **Eff<sup>E</sup>** se torna verdade no instante em que a ação termina (**end**).

A ação *fill(lake)* na missão de combate a incêndios implica que o VANT tenha água ao final da ação, **Eff<sup>E</sup>** =  $\{water\}$ . A ação *drop(forest)* implica que o VANT não tenha mais água ao final da ação e o foco de incêndio  $l \in \mathbf{l}_{forest}$  esteja apagado, **Eff<sup>E</sup>** =  $\{\neg water, \neg l\}$ . A ação *fly* não possui implicações de efeito discreto.

**Definição 13** (Equações de Dinâmica de uma Ação Durativa Híbrida). Considerando uma discretização do tempo  $t \in \{0, 1, 2, \dots, T\}$  em intervalos iguais a  $\Delta t$ , **Dyn** assume a forma típica das equações 6.1 e 6.2:

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \omega_t \quad \forall(t) \quad (6.1)$$

$$\mathbf{x}_t \in \mathbb{X}, \quad \mathbf{u}_t \in \mathbb{U}, \quad \omega_t \sim \mathcal{N}(0, \Sigma_{\omega_t}) \quad \forall(t) \quad (6.2)$$

onde  $\mathbf{x}_t \in \mathbb{X}$  e  $\mathbf{u}_t \in \mathbb{U}$  são restrições específicas, mantendo os estados e os controles dentro dos limites válidos para um dado VANT. A variável  $\omega_t$  é aleatória e independente de estado, representando distúrbios causados pela imprecisão na transição dos estados do

sistema. Assume-se que o distúrbio possa ser modelado como um ruído Gaussiano com média zero e matriz de covariância  $\Sigma_{w_t}$ . Na equação 6.1, as matrizes  $A$  e  $B$  definem a transição dos estados. Para o exemplo da missão de combate a incêndio, as equações de Dinâmica são dadas pelas expressões 6.3 e 6.4.

$$\begin{bmatrix} p^x \\ p^y \\ v^x \\ v^y \\ fuel \end{bmatrix}_{t+1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p^x \\ p^y \\ v^x \\ v^y \\ fuel \end{bmatrix}_t + \begin{bmatrix} \frac{\Delta t^2}{2} & 0 & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & -\Delta t \end{bmatrix} \begin{bmatrix} a^x \\ a^y \\ r_f \end{bmatrix}_t \quad (6.3)$$

$$\begin{aligned} \mathbf{x}_t \in \mathbb{X} &\Leftrightarrow \begin{bmatrix} (p^x, p^y) \in \mathbb{R}^2 \\ \sqrt{(v^x)^2 + (v^y)^2} \leq v^{max} \\ fuel \in [0, 100\%] \end{bmatrix} \\ \mathbf{u}_t \in \mathbb{U} &\Leftrightarrow \begin{bmatrix} \sqrt{(a^x)^2 + (a^y)^2} \leq a^{max} \\ r_f \geq 1 + R^f \sqrt{(a^x)^2 + (a^y)^2} \end{bmatrix} \end{aligned} \quad (6.4)$$

**Definição 14** (Duração de uma Ação Durativa Híbrida).  $\mathbf{d}$  é o tempo indicando a duração da ação com  $\mathbf{d} \in [lb, ub]$ , onde  $lb$  (*lower bound*) é o menor tempo e  $ub$  (*upper bound*) é o maior tempo dado para a ação ser executada.

No exemplo de combate a incêndios, a ação *fly* possui duração limitada em  $\mathbf{d} \in [0, +\infty]$ , a ação *fill*,  $\mathbf{d} \in [4, 8]$ , e a ação *drop*,  $\mathbf{d} \in [2, 5]$ .

**Definição 15** (Chance-Constrained Qualitative State Plan).  $\mathcal{P} = \langle \mathcal{E}, \mathcal{T}, \mathcal{A}, \mathcal{C} \rangle$ , onde:

- $\mathcal{E} = \{e_0, e_1, \dots\}$  é um conjunto de eventos discretos, cada evento  $e \in \mathcal{E}$  representa um instante no tempo.
- $\mathcal{A} = \{a_1, a_2, \dots\}$  é um conjunto de episódios, em que cada episódio  $a \in \mathcal{A}$  é definido entre dois eventos quaisquer e estabelecem a ordem desejada no cumprimento da missão, indicando as restrições que devem ser cumpridas no intervalo de tempo entre os eventos (*Def. 16*).
- $\mathcal{C} = \{c_1, c_2, \dots\}$  é o conjunto de *chance constraints* ( $c \in \mathcal{C}$ ), onde cada  $c$  limita o risco máximo  $\Delta_c$  que pode ser atribuído a um conjunto de episódios ( $\Psi_c$ , ver *Def. 16*).
- $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \dots\}$  são as restrições temporais entre os eventos, indicando os tempos mínimos ( $\mathcal{T}^{lb}$ ) e máximos ( $\mathcal{T}^{ub}$ ) de duração entre dois eventos quaisquer.

**Definição 16** (Episódios do  $\mathcal{P}$ ). Um episódio  $a = \langle e^S, e^E, \mathbf{Const}^S, \mathbf{Const}^R, \mathbf{Const}^E \rangle$  é composto por:



- $e^S$  um evento inicial do episódio.
- $e^E$  um evento final do episódio.
- $\text{Const}^S$  restrições de início (verdade no instante de início do episódio).
- $\text{Const}^R$  restrições de permanência (verdade em todo o tempo do episódio).
- $\text{Const}^E$  restrições de fim (verdade no último instante do episódio).

Assim, o HCCQSP para o exemplo considerado pode ser representado como ilustrado na Figura 23.

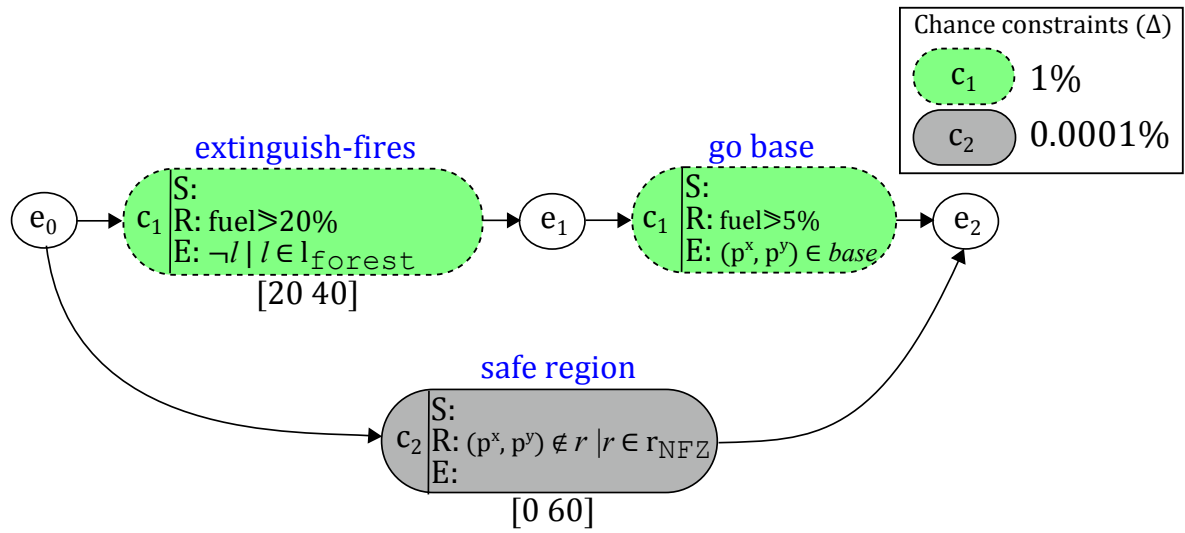


Figura 23 – O panejamento  $\mathcal{P}$  para a missão de combate a incêndios.

### 6.3 Modelo para o HCCQSP

Uma formulação matemática para o problema estabelecido será descrita a seguir. De forma geral, deseja-se minimizar uma função objetivo:

$$\text{Minimizar } \mathcal{J}(\cdot) \quad (6.5)$$

sujeito às condições iniciais para os estados  $\mathbf{x}_t$  e os literais  $\mathbf{l}_t$ .

$$\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_0, \Sigma_{x_0}) \quad (6.6)$$

$$\mathbf{l}_0 = \hat{\mathbf{l}}_0 \quad (6.7)$$

### 6.3.1 Restrições modelando o comportamento das ações

Dada a definição das ações, temos que uma única ação é executada por vez (ou a cada passo). Logo, executar um fluxo de ações em sequência para o cumprimento de uma missão é o mesmo que percorrer um caminho em um grafo conforme apresentado em (LI, 2010). As arestas desse grafo representam as ligações entre as ações possíveis de serem executadas, por exemplo, na missão apresentada na Figura 22, é preciso primeiro executar a ação de voar *fly* até o lago para executar a ação pegar água *fill*. A Figura 24 apresenta um exemplo do fluxo das ações a cada passo  $n$  para a missão de combate a incêndios da Figura 22. As linhas tracejadas indicam todas as aresta factíveis para a missão e o caminho contínuo em vermelho apresenta o fluxo seguido na solução do exemplo da Figura 22.

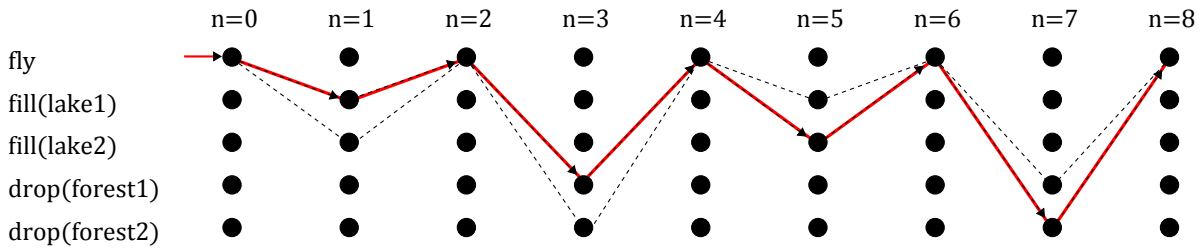


Figura 24 – Fluxo das ações para o cumprimento da missão de combate a incêndios.

A variável  $H_{nh} \in \{0, 1\}$  indica se a ação  $h$  é executada no passo  $n$ , assim a restrição (6.8) garante que no máximo uma ação  $h$  pode ser executada a cada passo  $n$ .

$$\sum_h H_{nh} \leq 1 \quad \forall(n) \quad (6.8)$$

A restrição (6.9) exclui as ações inexistentes no grafo:

$$H_{nh} = 0 \quad \forall(n, h \mid \nexists(n, h)) \quad (6.9)$$

A variável  $D_{nh} \in \mathbb{Z}^+$  é utilizada para calcular a duração da ação  $h$  no passo  $n$  pela restrição 6.10.

$$\mathbf{d}_h^{\text{lb}} \cdot H_{nh} \leq D_{nh} \cdot \Delta t \leq \mathbf{d}_h^{\text{ub}} \cdot H_{nh} \quad \forall(n, h) \quad (6.10)$$

O tempo a cada passo  $n$  é dado por  $T_n \geq 0$ , onde  $T_{n^\uparrow}$  representa o tempo no próximo passo ( $n^\uparrow \Leftrightarrow n + 1$ ) conforme descrito na restrição 6.11.

$$T_{n^\uparrow} = T_n + \sum_h D_{nh} \cdot \Delta t \quad \forall(n) \quad (6.11)$$

O *scheduling* ( $Z_{ht} \in \{0, 1\}$ ) das ações  $h$  executadas a cada instante de tempo  $t$  é estabelecido pelas restrições 6.12 e 6.13.

$$[ T_n \leq t \cdot \Delta t \leq T_{n\uparrow} ] \wedge H_{nh} \Rightarrow Z_{ht} \quad \forall(n, h, t) \quad (6.12)$$

$$[ T_N < t \cdot \Delta t ] \Rightarrow \neg Z_{ht} \quad \forall(h, t) \quad (6.13)$$

A restrição 6.14 estabelece que uma única ação  $h$  pode ser executada no máximo a cada instante  $t$ .

$$\sum_h Z_{ht} \leq 1 \quad \forall(t) \quad (6.14)$$

As condições iniciais (**start** ou **S**) para a ação  $h$  ser executada são definidas nas restrições 6.15 e 6.16, onde  $Z_{ht}^S \geq 0$  é verdade no instante em que a ação  $h$  é iniciada.

$$Z_{ht}^S \geq Z_{ht} - Z_{ht\downarrow} \quad \forall(h, t) \quad (6.15)$$

$$Z_{ht}^S \Rightarrow \mathbf{Cond}_{ht}^S \quad \forall(h, t) \quad (6.16)$$

Condições finais (**end** ou **E**) para a ação  $h$  ser executada também são definidas pelas restrições 6.17 e 6.18, onde  $Z_{ht}^E \geq 0$  é verdade no instante em que a ação  $h$  é terminada.

$$Z_{ht}^E \geq Z_{ht} - Z_{ht\uparrow} \quad \forall(h, t) \quad (6.17)$$

$$Z_{ht}^E \Rightarrow \mathbf{Cond}_{ht}^E \quad \forall(h, t) \quad (6.18)$$

Temos condições de permanência (**remain** ou **R**) para a ação  $h$  ocorrer definida nas restrições 6.19 e 6.20, onde  $Z_{ht}^R \geq 0$  é verdade durante todo instante  $t$  em que a ação  $h$  é executada.

$$Z_{ht}^R = Z_{ht} \quad \forall(h, t) \quad (6.19)$$

$$Z_{ht}^R \Rightarrow \mathbf{Cond}_{ht}^R \quad \forall(h, t) \quad (6.20)$$

A dinâmica (**Dyn**) da ação  $h$  é aplicada a todo instante  $t$  em que a ação é executada seguindo a restrição 6.21.

$$Z_{ht} \Rightarrow \mathbf{Dyn}_{ht} \quad \forall(h, t) \quad (6.21)$$

Efeitos (**Eff**) discretos iniciais (**start** ou **S**) da ação  $h$  são estabelecidos nas restrições 6.22 - 6.24, onde  $l_t$  é estado do literal a cada instante de tempo  $t$ .

$$Z_{ht}^S \Rightarrow [l_t = \hat{l}] \quad \forall(h, t, l \in \mathbf{Eff}_h^S) \quad (6.22)$$

$$Z_{ht}^S \Rightarrow [l_t = l_t^\downarrow] \quad \forall(h, t, l \notin \mathbf{Eff}_h^S) \quad (6.23)$$

$$\neg \left[ \bigvee_h Z_{ht}^S \right] \Rightarrow [l_t = l_t^\downarrow] \quad \forall(t, l) \quad (6.24)$$

Efeitos (**Eff**) discretos finais (**end** ou **E**) da ação  $h$  também ocorrem pelas restrições 6.25-6.27.

$$Z_{ht}^E \Rightarrow [l_t = \hat{l}] \quad \forall(h, t, l \in \mathbf{Eff}_h^E) \quad (6.25)$$

$$Z_{ht}^E \Rightarrow [l_t = l_t^\downarrow] \quad \forall(h, t, l \notin \mathbf{Eff}_h^E) \quad (6.26)$$

$$\neg \left[ \bigvee_h Z_{ht}^E \right] \Rightarrow [l_t = l_t^\downarrow] \quad \forall(t, l) \quad (6.27)$$

Uma ação  $h$ , no máximo, pode ser iniciada e terminada a cada instante  $t$  seguindo as restrições 6.28 e 6.29.

$$\sum_h Z_{ht}^S \leq 1 \quad \forall(t) \quad (6.28)$$

$$\sum_h Z_{ht}^E \leq 1 \quad \forall(t) \quad (6.29)$$

### 6.3.2 Restrições modelando o planejamento da missão

O tempo em que o evento ( $e$ ) ocorre é dado por  $T_e$ , onde  $K_e \in \mathbb{Z}^+$  calcula o tempo decorrido entre o evento atual ( $e$ ) e o próximo ( $e^\uparrow$ ) na restrição 6.30. Considera-se  $T_{e_0} = 0$  para o evento inicial ( $e_0$ ).

$$T_{e^\uparrow} = T_e + K_e \cdot \Delta t \quad \forall(e) \quad (6.30)$$

As restrições de tempo 6.31-6.34 dos episódios  $a \in \mathcal{A}$  precisam ser obedecidas.

$$\mathcal{T}_a^{lb} \leq T_{e_a^E} - T_{e_a^S} \leq \mathcal{T}_a^{ub} \quad \forall (a \in \mathcal{A}) \quad (6.31)$$

$$T_{e_a^S} = t \cdot \Delta t \quad \Rightarrow \quad Y_{st} \quad \forall (t, a \in \mathcal{A}, s \in \mathbf{Const}_a^S) \quad (6.32)$$

$$T_{e_a^E} = t \cdot \Delta t \quad \Rightarrow \quad Y_{st} \quad \forall (t, a \in \mathcal{A}, s \in \mathbf{Const}_a^E) \quad (6.33)$$

$$T_{e_a^S} \leq t \cdot \Delta t \leq T_{e_a^E} \quad \Rightarrow \quad Y_{st} \quad \forall (t, a \in \mathcal{A}, s \in \mathbf{Const}_a^R) \quad (6.34)$$

A probabilidade de sucesso das restrições da missão *chance-constraints* precisam ser garantidas conforme restrição 6.35 que especifica o limite de risco de falha dado por  $\Delta_c$ .

$$Pr \left[ \bigwedge_{a \in \Psi_c} \bigwedge_{s \in \mathbf{Const}_a^X} \bigwedge_t (Y_{st} \Rightarrow \mathbf{x}_t \in R_s) \right] \geq 1 - \Delta_c \quad \forall (c) \quad (6.35)$$

Por fim, é preciso garantir também que foram cumpridas as tarefas envolvendo os literais, equação 6.36, conforme as tarefas especificadas nas restrições na missão.

$$Y_{st} \quad \Rightarrow \quad \bigwedge_{l \in \mathbf{Const}_s^1} [l_t = \hat{l}] \quad \forall (t, a \in \mathcal{A}, s \in \mathbf{Const}_a^1) \quad (6.36)$$

Os limitantes das variáveis são:

$$\mathbf{x}_t \in \mathbb{X} \subseteq \mathbb{R}^n, \mathbf{u}_t \in \mathbb{U} \subseteq \mathbb{R}^m, \mathbf{l}_t \in \{0, 1\}^k \quad (6.37)$$

$$(T_n, T_e) \geq 0 \quad (6.38)$$

$$(D_{nh}, k_e) \in \mathbb{Z}_+ \quad (6.39)$$

$$(H_{nh}, Z_{ht}, Y_{st}) \in \{0, 1\} \quad (6.40)$$



---

## RESULTADOS

---

Neste capítulo serão sumarizados os principais resultados alcançados durante esta pesquisa. Maiores detalhes podem ser encontrados nos artigos publicados, submetidos ou em fase final de elaboração:

### 1. Artigos aceitos/publicados:

- **Anexo A:** (ARANTES *et al.*, 2016a) apresenta contribuições relevantes ao reformular o CCPP, descrito na literatura como um Problema Não-Linear Inteiro Misto (PNLIM) e aqui reformulado como um Problema Linear Inteiro Misto (PLIM). Além disso, foi possível avançar nos aspectos de escalabilidade e representação. Uma escalabilidade melhor foi alcançada com a proposição de um novo método *Hybrid Incremental Solution Approach* (HISA), que é capaz de resolver cenários com maior número de obstáculos e maior quantidade de waypoints. Os avanços na representação permitiram computar as incertezas que existiam entre os pontos da rota, aspecto não tratado em trabalhos anteriores. Os principais resultados alcançados nesse trabalho são sumarizados nas seções 7.1.1 até 7.1.4.
- **Anexo C:** (ARANTES *et al.*, 2016b) explorou o uso das técnicas de computação evolutiva para resolver o problema, onde os resultados se mostraram promissores ao retornar soluções de boa qualidade em um reduzido tempo computacional. Trata-se de um aspecto relevante, caso o planejador de rota venha a fazer parte do sistema embarcado na aeronave. A seção 7.1.6 sumariza este resultado.
- **Anexo H:** (ARANTES *et al.*, 2017b) apresenta uma arquitetura de hardware para VANTS na qual foram embarcados os algoritmos de computação evolutiva propostos para o problema. Detalhes a respeito serão apresentados na seção 7.1.7.

- **Anexo F:** (ARANTES *et al.*, 2015), este trabalho explorou o uso da computação evolutiva para um problema de pouso emergencial, o trabalho encontra-se em anexo, mas seus resultados não serão apresentados aqui.
- **Anexo G:** (ARANTES *et al.*, 2017a), trata-se de uma extensão para revista do trabalho anterior.
- **Anexo E:** (ARIU *et al.*, 2017), este trabalho apresentou o uso do princípio reflexivo como uma alternativa melhor para o tratamento no desvio de obstáculos.

## 2. Artigos em elaboração:

- **Anexo B:** (ARANTES *et al.*, 2017d) contempla os principais resultados alcançados com escalabilidade e representação para o CCQSP. Os principais resultados estão descritos na seção 7.2.
- **Anexo D:** (ARANTES *et al.*, 2017c) unifica o uso do princípio reflexivo com o tratamento contínuo da dinâmica da aeronave, garantindo uma matemática mais completa e avançando no aspecto de representação para tratamento das colisões. Os detalhes estão apresentados na seção 7.1.5.

## 7.1 Resultados para o CCPP

Nesta seção são reportados os resultados computacionais obtidos ao solucionar o planejamento de missão para o CCPP a partir de três abordagens: um método exato ( $CNPP^{LEN}$ ) e duas heurísticas determinísticas (CSA e HISA).

### 7.1.1 Metodologia e cenários

Os três métodos utilizados para solucionar o CCPP e os cenários gerados para o planejamento da missão serão descritos. O método  $CNPP^{LEN}$  se trata da resolução exata, através do solver CPLEX, do modelo linear inteiro misto proposto neste trabalho para o CCPP. Os resultados obtidos pelo  $CNPP^{LEN}$  serviram como parâmetro para comparar o desempenho das heurísticas Customized Solution Approach (CSA) e Hybrid Incremental Solution Approach (HISA). CSA é uma heurística proposta em (BLACKMORE; ONO; WILLIAMS, 2011) que fixa valores para o risco a ser alocado de forma a se evitar a resolução de um modelo não-linear durante a alocação do risco. HISA é uma heurística proposta no decorrer desta tese e descrita em detalhes no artigo do anexo A. Essa heurística utiliza um grafo de visibilidade construído de forma a respeitar uma margem mínima de segurança para se evitar colisões com obstáculo. Um caminho mais curto é determinado usando o algoritmo de Dijkstra a partir do qual os riscos são alocados de forma incremental.



Os cenários estabelecidos para o CCPP consideram dois grupos de mapas: mapas 2D e 3D. Para cada grupo, dois outros são definidos como regular e non-regular. Para os mapas 2D, os obstáculos não tem altura definida, logo o VANT não pode voar sobre estes sendo considerados como zonas de exclusão aérea (non-fly zones - NFZ). Os mapas 3D apresentam NFZ, como também consideram obstáculos com alturas definidas que o VANT pode sobrevoar. Os mapas do tipo non-regular permitem sobreposição, gerando regiões não regulares que se comportam como obstáculos não convexos. Os mapas regulares são gerados sem sobreposição o que leva a uma maior ocupação do mapa. A Figura 25 exemplifica cada um dos 4 grupos de mapas considerados.

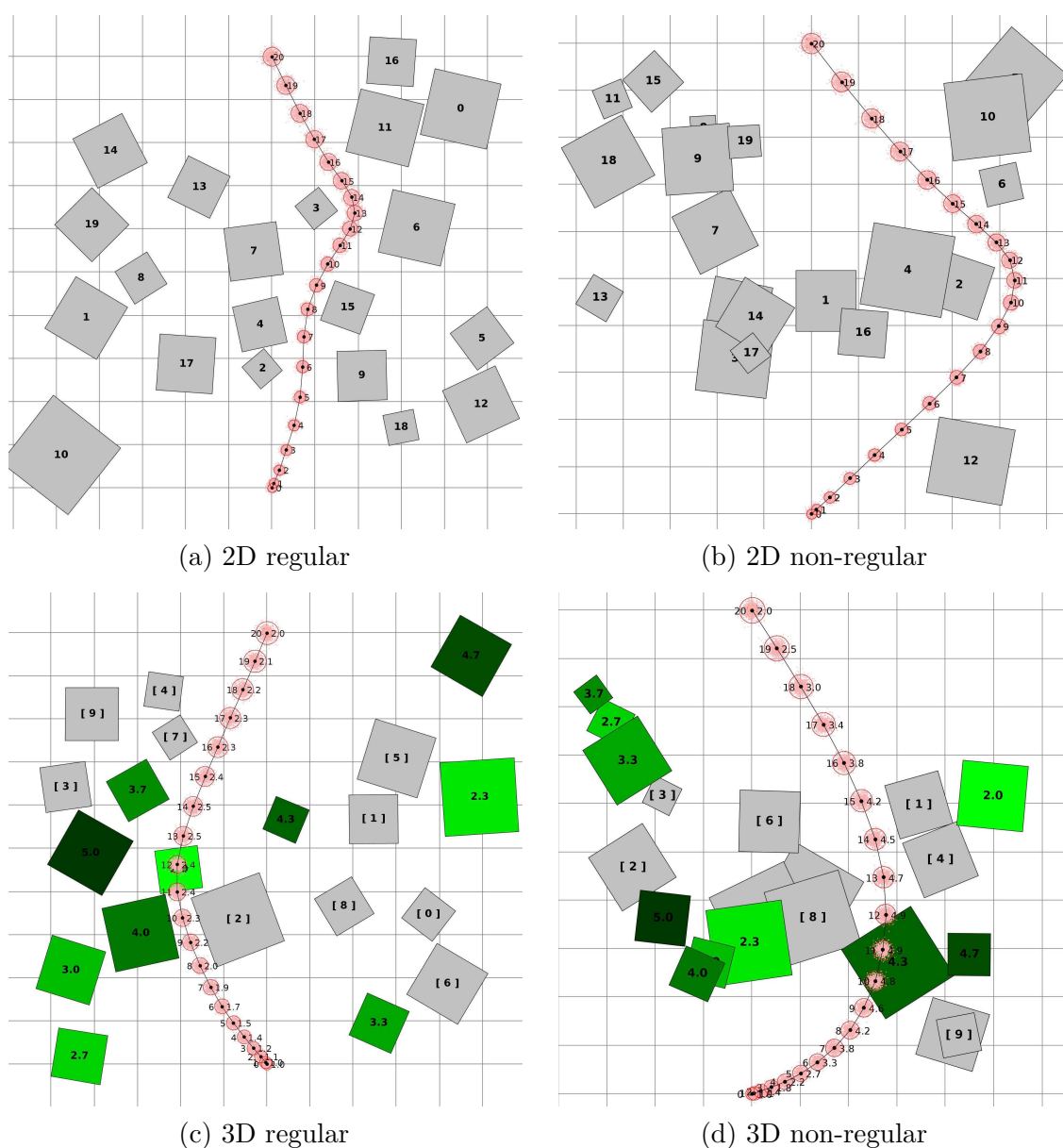


Figura 25 – Exemplos de mapas 2D, 3D, regular e non-regular

Um gerador de mapas foi desenvolvido com dois grandes conjuntos de instâncias aleatoriamente geradas. No primeiro conjunto, são definidos grupos baseados no crescimento

do número de obstáculos  $J = 12, 16, 20, 24, 28, 32, 36, 40$  com  $T = 20$  e  $\Delta = 0.001$ . Um total de 50 mapas são criados para cada valor de  $J$ , totalizando assim 1600 mapas para 2D, 3D, regular e non-regular. No segundo conjunto, um total de 50 mapas são gerados baseados no crescimento do horizonte de tempo  $T = 10, 15, 20, 25, 30, 35, 40, 45$  com  $J = 20$  e  $\Delta = 0.001$ . Também um total de 1600 mapas são definidos. Logo, há um total de 3000 diferentes mapas, visto que há uma sobreposição para  $J = 20$  e  $T = 20$ .

Os métodos foram avaliados nestes mapas considerando três critérios: soluções factíveis, tempo computacional e distância percorrida. Esses critérios foram estabelecidos já que se faz necessário ter bons resultados em todos eles para viabilizar aplicações reais. Por exemplo, se um método tem problemas em encontrar soluções factíveis (rotas), então haverá uma complicação em voo visto que um VANT não poderá voar sem uma rota. O método precisa encontrar rotas com reduzido tempo computacional, uma vez que durante a execução da missão, a aeronave não pode esperar por replanejamentos demorados na estação de solo ou mesmo em um sistema embarcado. A distância percorrida pela rota é importante visto que longas distâncias refletem um maior consumo de combustível ou mesmo violam a duração da missão planejada.

### 7.1.2 Resultados de escalabilidade para mapas 2D

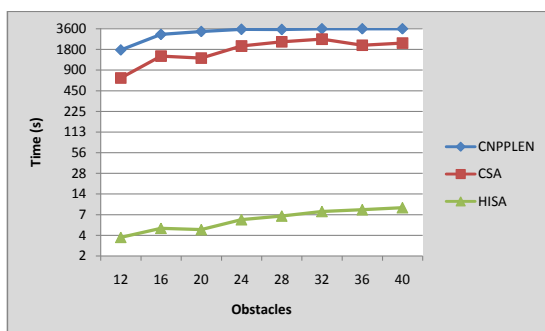
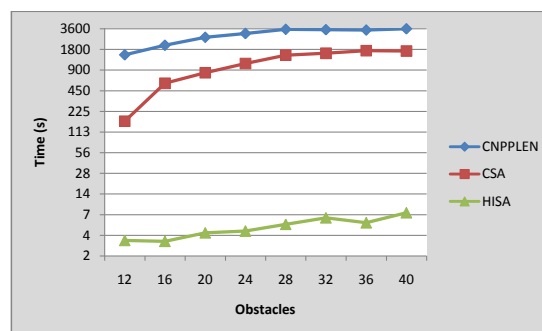
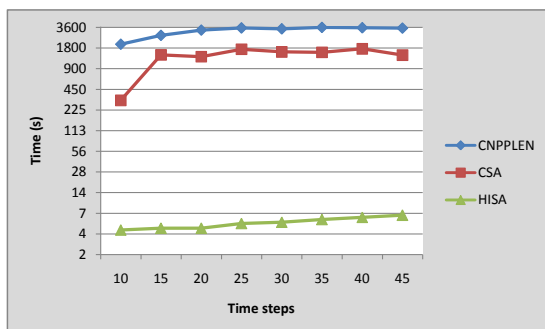
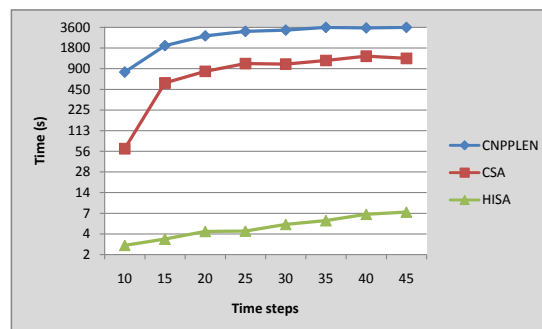
A Tabela 11 apresenta o número de soluções factíveis retornado por cada método nos mapas 2D regular e non-regular aos se incrementar  $J$  e  $T$ . A linha *same maps* considera o conjunto comum de mapas nos quais todos os métodos encontraram soluções. O HISA retornou soluções em todos os conjuntos, exceto em um regular e outro non-regular com  $T = 10$ . O CSA se mostrou competitivo, enquanto o método exato ( $CNPP^{LEN}$ ) teve problemas em encontrar soluções quando  $J$  e  $T$  aumentaram seus valores. O número de obstáculos e horizonte de planejamento tem impacto considerável no número de variáveis de decisão do  $CNPP^{LEN}$ , o que torna difícil a aplicação do método de enumeração.

As Figuras 26a-26d apresentam a média do tempo computacional. Para uma comparação justa, são considerados apenas os mapas comuns (*same maps*). O HISA é a abordagem mais rápida ao consumir em média menos que 9 segundos para mapas regular e 7.5 segundos para mapas non-regular (Figuras 26a e 26b). Por outro lado,  $CNPP^{LEN}$  leva mais de 1700 segundos para mapas regular e 1500 segundos para mapas non-regular. O CSA tem uma performance melhor que o  $CNPP^{LEN}$ , mas pior que o HISA. Nas Figuras 26c e 26d, o  $CNPP^{LEN}$  também consome mais tempo quando  $T$  aumenta. No geral, o HISA é o método mais rápido ao gastar menos que 6.6 segundos em média para todos os mapas regulares e menos que 7.3 segundos para todos os mapas non-regular.

As Figuras 27a-27d comparam a média da distância que precisa ser percorrida pelo VANT para o cumprimento da missão. Novamente, são consideradas apenas as rotas finais retornadas nos mapas onde todos os métodos encontraram soluções (*same*

Tabela 11 – Número de soluções factíveis para mapas 2D com aumento de  $J$  e  $T$ 

Obstacles ( $J$ )	12	16	20	24	28	32	36	40
<b>Regular</b>								
$CNPP^{LEN}$	48	50	48	45	47	34	29	18
$CSA$	50	50	50	50	50	48	48	47
$HISA$	50	50	50	50	50	50	50	50
same maps	48	50	48	45	47	33	27	17
<b>Non-regular</b>								
$CNPP^{LEN}$	49	50	50	47	47	44	44	36
$CSA$	50	50	50	50	50	50	50	50
$HISA$	50	50	50	50	50	50	50	50
same maps	49	50	50	47	47	44	44	36
<b>Time Steps (<math>T</math>)</b>								
<b>Regular</b>								
$CNPP^{LEN}$	49	48	48	47	45	37	33	29
$CSA$	49	49	50	50	50	49	50	47
$HISA$	49	50	50	50	50	50	50	50
same maps	47	47	48	47	45	37	33	28
<b>Non-regular</b>								
$CNPP^{LEN}$	49	50	50	47	45	45	44	44
$CSA$	50	50	50	50	50	50	50	50
$HISA$	49	50	50	50	50	50	50	50
same maps	48	50	50	47	45	45	44	44

(a) Aumento de  $J$ , mapas 2D regular(b) Aumento de  $J$ , mapas 2D non-regular(c) Aumento de  $T$ , mapas 2D regular(d) Aumento de  $T$ , mapas 2D non-regularFigura 26 – Tempo computacional médio quando aumenta-se  $J$  e  $T$  em mapas 2D

$maps$ ). O VANT sempre inicia do ponto  $(0,0)$  e o destino final é o ponto  $(0,-10)$ , assim é esperado que a distância percorrida seja um valor entre 10 e 20 unidades. As distâncias percorridas são maiores para mapas regular do que mapas non-regular quando  $J$  aumenta, visto que seus obstáculos estão mais espalhados pelo mapa. A medida que  $T$  aumenta, é possível encontrar caminhos um pouco melhores. Isso faz sentido, uma vez que implica uma

melhor discretização do horizonte de planeamento, possibilitando um melhor planeamento. Apesar de todas as abordagens terem caminho de tamanho similares, o HISA encontrou em média os caminhos mais curtos para a maioria dos resultados.

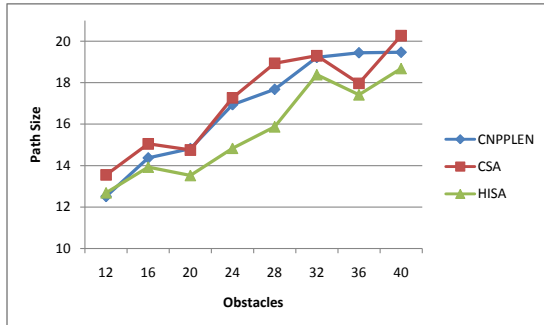
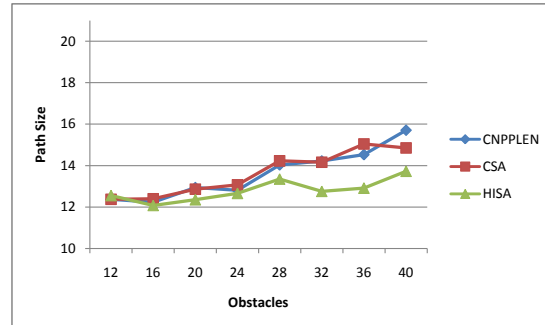
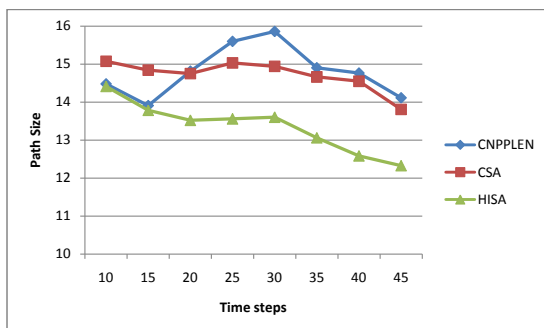
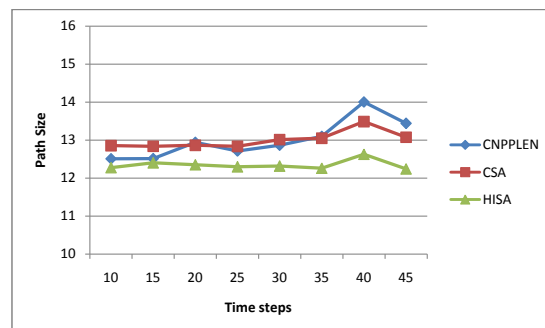
(a) Aumento de  $J$ , mapas 2D regular(b) Aumento de  $J$ , mapas 2D non-regular(c) Aumento de  $T$ , mapas 2D regular(d) Aumento de  $T$ , mapas 2D non-regular

Figura 27 – Distância percorrida média quando aumenta-se  $J$  e  $T$  em mapas 2D

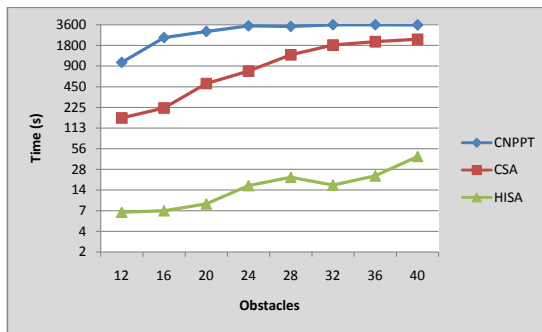
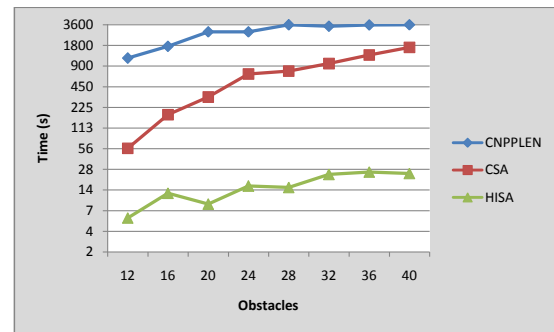
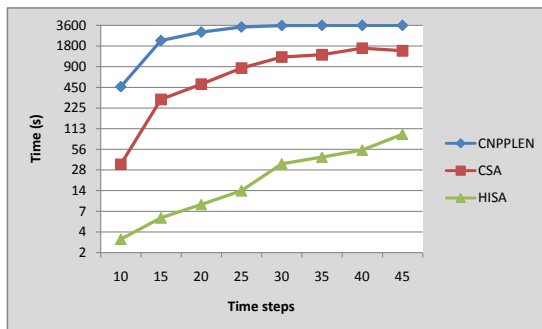
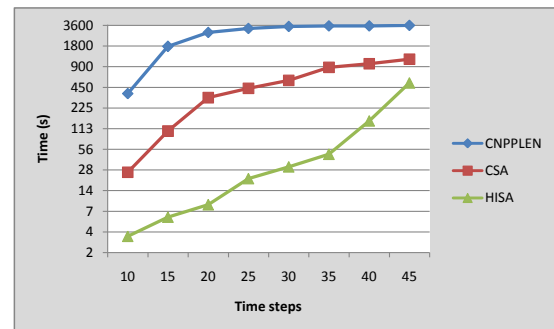
Resumindo, o CSA é capaz de retornar soluções factíveis com o incremento de  $J$  e  $T$ , mas consome um tempo computacional considerável. O  $CNPP^{LEN}$  tem problemas em encontrar soluções factíveis, dentro de 1 hora de tempo limite, quando  $J$  e  $T$  aumentam. O HISA aparece como a melhor abordagem, visto que resolve a maioria dos mapas, sendo computacionalmente rápido e encontrando rotas competitivas em termos de distância percorrida.

### 7.1.3 Resultados de escalabilidade para mapas 3D

A Tabela 12 apresenta o número de soluções factíveis retornado pelos métodos para mapas 3D regular e non-regular quando é incrementado os valores de  $J$  e  $T$ . O número de *same maps* também é apresentado. O HISA e CSA conseguem soluções factíveis para todos os grupos, exceto CSA para um mapa non-regular com  $J = 36$ . O  $CNPP^{LEN}$  não encontrou soluções em 19 mapas para regular e non-regular maps ao aumentar o valor de  $J$ , também falha em 11 mapas regular e 12 mapas non-regular ao aumentar  $T$ . As Figuras 28a-28d mostram o tempo computacional médio para os mapas com soluções em comum. O HISA novamente é o método mais rápido.  $CNPP^{LEN}$  consome mais de 1800 segundos na maioria dos mapas. O CSA é melhor que o  $CNPP^{LEN}$  e pior que o HISA.

Tabela 12 – Número de soluções factíveis para mapas 3D com o aumento de  $J$  e  $T$ 

Obstacles ( $J$ )	12	16	20	24	28	32	36	40
<b>Regular</b>								
$CNPP^{LEN}$	49	49	48	50	45	45	50	45
$CSA$	50	50	50	50	50	50	50	50
$HISA$	50	50	50	50	50	50	50	50
same maps	49	49	48	50	45	45	50	45
<b>Non-regular</b>								
$CNPP^{LEN}$	50	50	48	47	46	46	50	44
$CSA$	50	50	50	50	50	50	49	50
$HISA$	50	50	50	50	50	50	50	50
same maps	50	50	48	47	46	46	49	44
<b>Time Steps (<math>T</math>)</b>								
<b>Regular</b>								
$CNPP^{LEN}$	49	47	48	48	49	49	49	50
$CSA$	50	50	50	50	50	50	50	50
$HISA$	50	50	50	50	50	50	50	50
same maps	49	47	48	48	49	49	49	50
<b>Non-regular</b>								
$CNPP^{LEN}$	50	48	48	50	50	50	46	46
$CSA$	50	50	50	50	50	50	50	50
$HISA$	50	50	50	50	50	50	50	50
same maps	50	48	48	50	50	50	46	46

(a) Aumento de  $J$ , mapas 3D regular(b) Aumento de  $J$ , mapas 3D non-regular(c) Aumento de  $T$ , mapas 3D regular(d) Aumento de  $T$ , mapas 3D non-regularFigura 28 – Tempo computacional médio quando aumenta-se  $J$  e  $T$  em mapas 3D

O HISA enfrenta mais problemas em resolver mapas 3D do que 2D, uma vez que para mapas 2D seu tempo computacional médio era menor que 8 segundos. Este comportamento é esperado já que o HISA dá ênfase à otimização dos obstáculos que não possuem altura definida, representados como NFZ, enquanto os obstáculos com alturas definidas são deixados para serem resolvidos como submodelos do tipo PLIM. O CSA

consome em média a metade do tempo do  $CNPP^{LEN}$ . Trata-se de um comportamento esperado considerando como foi desenvolvida a heurística proposta por (BLACKMORE; ONO; WILLIAMS, 2011). CSA divide a busca em duas grandes etapas, nas quais é dado a metade do tempo limite para cada etapa. As Figuras 29a até 29d mostram a distância percorrida, onde não há grandes diferenças entre os métodos.

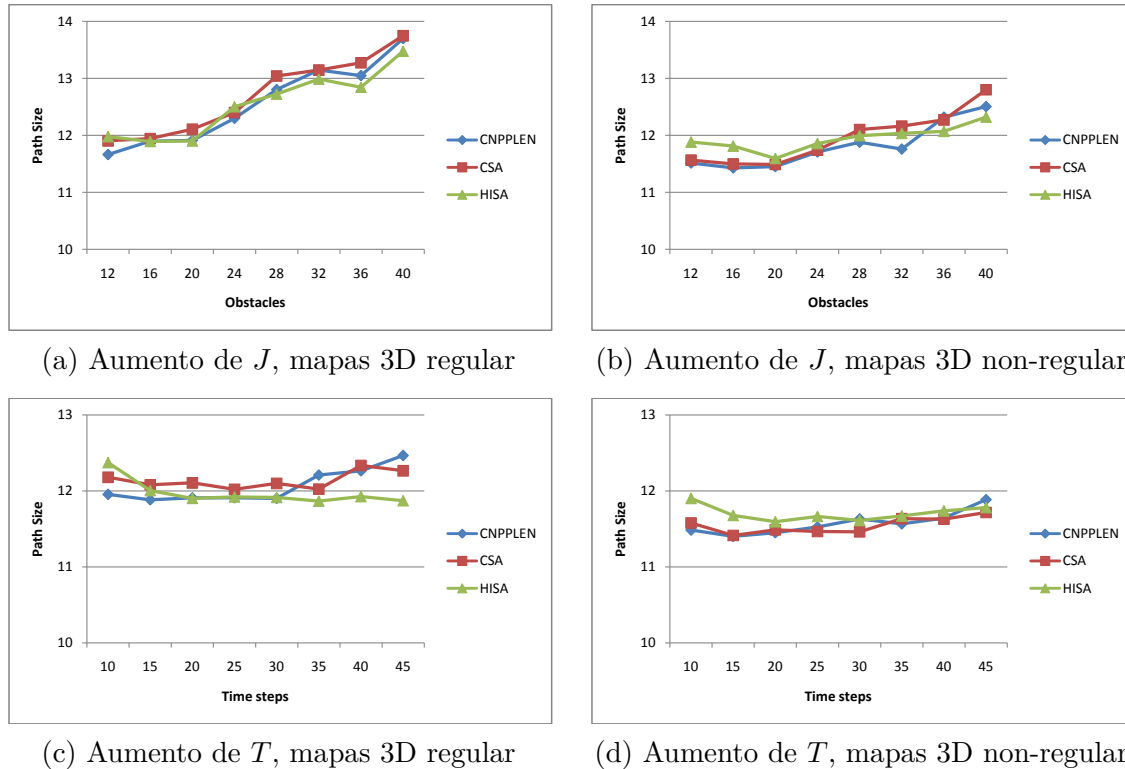


Figura 29 – Distância percorrida média quando aumenta-se  $J$  e  $T$  em mapas 3D

Os resultados para mapas 3D indicam que todos os métodos são capazes de encontrar soluções factíveis, o que pode ser explicado pelo fato de um espaço 3D facilitar o voo ao tornar possível sobrevoar alguns obstáculos. O HISA é novamente a melhor abordagem, conseguindo soluções em um pequeno tempo computacional e com qualidade similar quando comparadas as soluções dos outros métodos. O CSA é competitivo contra o HISA em termos de soluções factíveis, mas permanece gastando maior tempo computacional também em mapas 3D.

#### 7.1.4 Resultados em simulação com FlightGear

Simulações foram executadas para ilustrar o comportamento da aeronave ao seguir as rotas definidas pelo HISA. O simulador FlightGear (FG) foi escolhido por ser uma iniciativa de código aberto e com vários recursos disponíveis e bem documentados. O VANT simulado utiliza a dinâmica do Cessna 172, mas a aeronave não é conduzida por uma pessoa em nenhum momento. Um piloto automático foi desenvolvido para controlar o

voou da aeronave e seguir a rota planejada. A Figura 30 mostra a tela principal do ambiente desenvolvido para integrar o FG, piloto automático e execução do HISA. O HISA envia uma rota ao piloto automático, que executa a rota passando pelas regiões indicadas na missão.

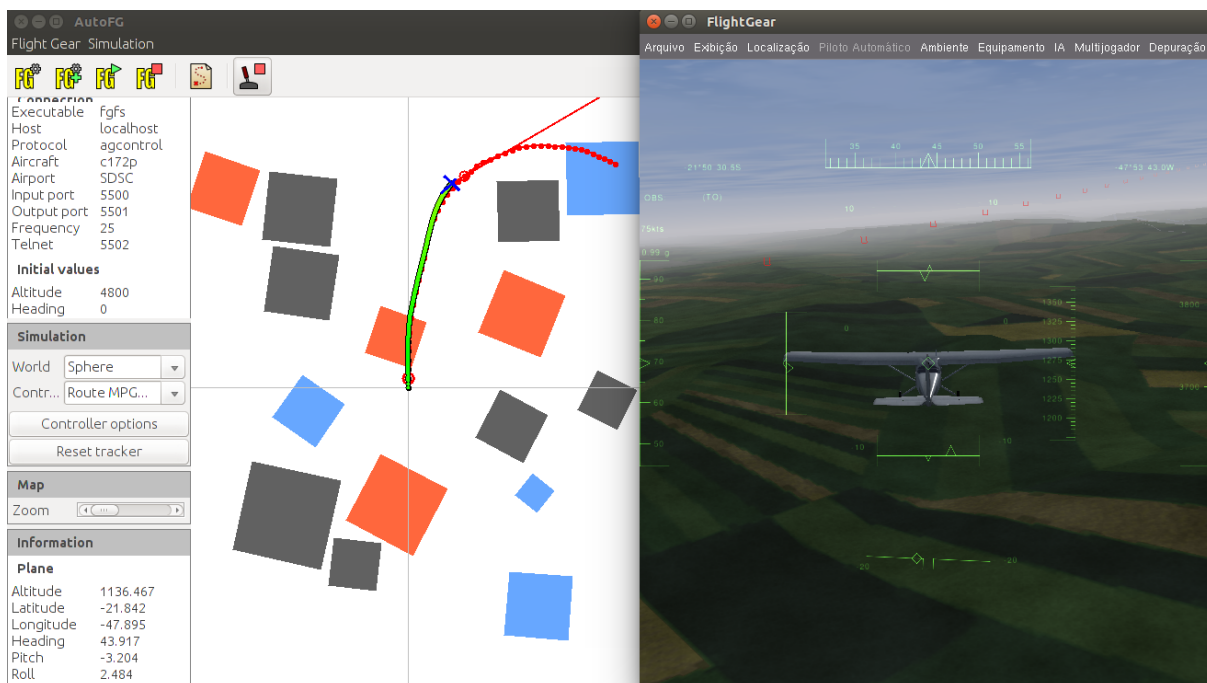


Figura 30 – Integração do Flightgear, HISA e piloto automático.

Um cenário foi previamente estabelecido para o teste simulado que inclui um aeroporto a partir do qual a aeronave decola e pousa. Um total de 4 tipos de regiões para cumprimento de missão (A, B, C e D) foi definido, onde a aeronave deve voar sem colidir com seis obstáculos mapeados (polígonos em cinza) como ilustrado na Figura 31. A aeronave deve ir do aeroporto até a região A e depois voar  $A \rightarrow B \rightarrow C \rightarrow D$ , terminando a missão ao retornar para o aeroporto.

Um total de 100 simulações foi executado, considerando as configurações padrões do FG (velocidade e direção randômica dos ventos). As incertezas são configuradas como  $\sigma_{x_0} = 30m$  e  $\sigma_{w_t} = 10m$ , baseados em testes preliminares com o Cessna 172. Um risco total de  $\Delta = 0.001$  foi estabelecido para o cumprimento de todas as missões. As 100 rotas são ilustradas na Figura 31, onde é possível ver que o desvio dos obstáculos foi satisfatório dado o nível de risco assumido.

A margem de segurança pode ser melhor avaliada na Figura 32. O HISA definiu caminhos com uma distância mínima esperada de 250.1 metros (linha vermelha). A Figura 32 apresenta a distância mínima, no qual a aeronave chegou próximo de colidir com qualquer dos obstáculos durante cada simulação. No pior caso, a aeronave chegou a 188.9 metros de proximidade com um obstáculo na simulação 91. Estes resultados indicam que as rotas projetadas pelo HISA são conservadoras o suficiente para evitar colisão, mantendo



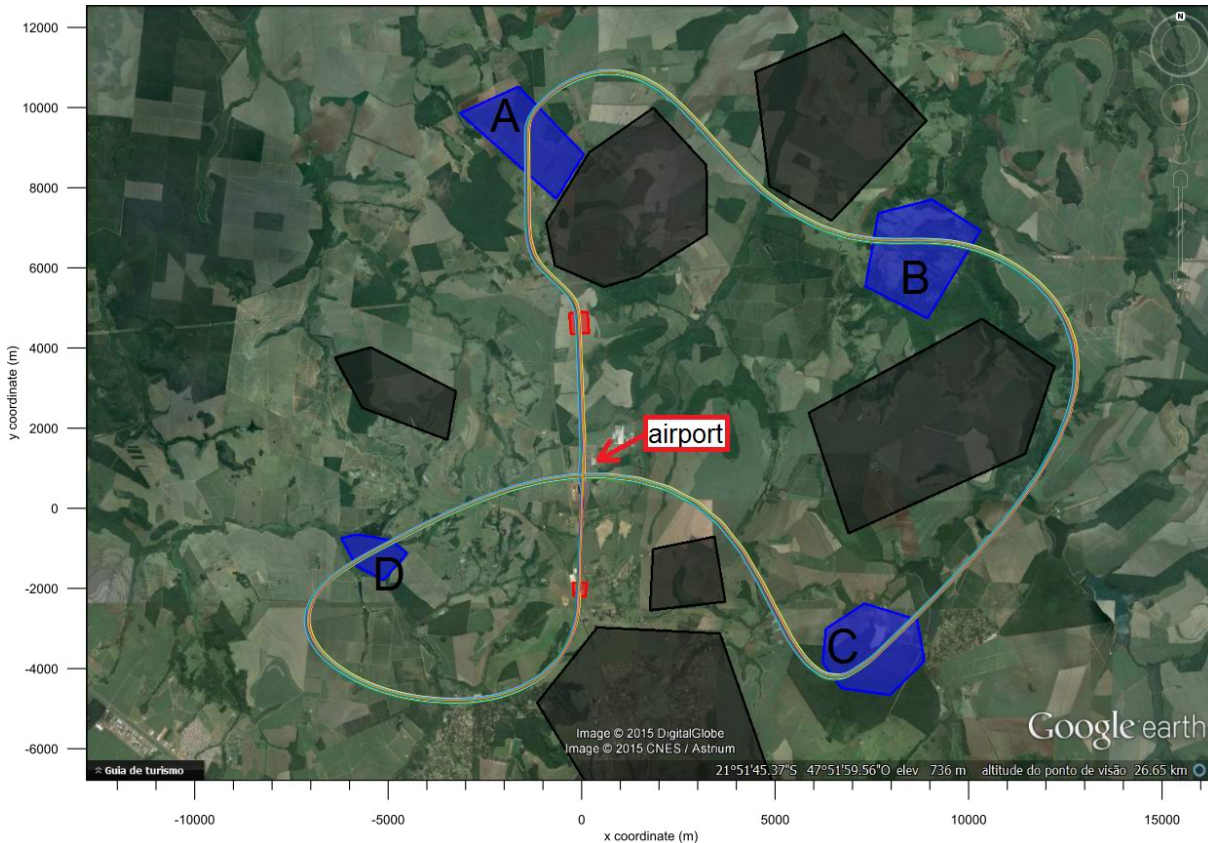


Figura 31 – Simulação no FlightGear com 100 rotas encontradas pelo HISA.

uma distância de segurança razoável mesmo com perturbações externas e internas. Há um vídeo disponível online para melhor visualização de uma das simulações executadas<sup>1</sup>.

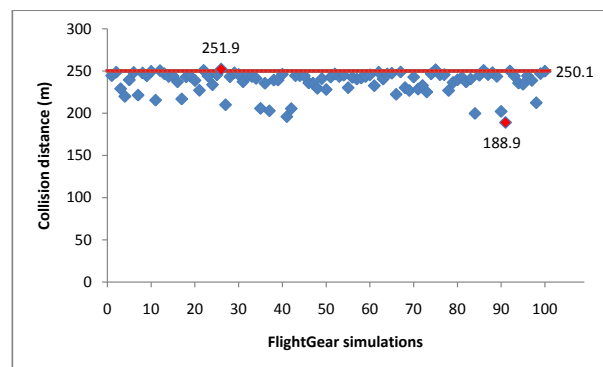


Figura 32 – Minimum collision distance considering all obstacles.

### 7.1.5 Avaliando o desvio de obstáculos

Uma das contribuições de melhoria na representação do problema proposta por esta tese está no tratamento das colisões. Nos trabalhos considerados na literatura, os riscos de colisão eram computados apenas nos pontos discretos da trajetória como descrito em

<sup>1</sup> <<https://youtu.be/i8flhA4yf90>>



(BLACKMORE; ONO; WILLIAMS, 2011). Assim, a primeira contribuição no tratamento das colisões considerava a região convexa definida pela alocação dos riscos entre dois pontos quaisquer da trajetória, resultando no trabalho publicado e presente no Anexo A. Outra contribuição foi o uso do princípio reflexivo que propõe uma matemática mais simples de ser implementada em um PLIM, além de fazer um tratamento mais realiza em relação ao crescimento da incerteza. Isso resultou em um trabalho em colaboração com a equipe do prof. Brian Williams do CSAIL/MIT, publicado na *Thirty-First AAAI Conference on Artificial Intelligence* (AAAI 2017), Anexo E. Por fim, uma terceira contribuição adicionou também o tratamento da dinâmica contínua entre dois pontos quaisquer da rota junto com o princípio reflexivo, resultando no trabalho ainda não publicado, apresentado no Anexo D. Todos esses avanços são resumidos na Figura 33.

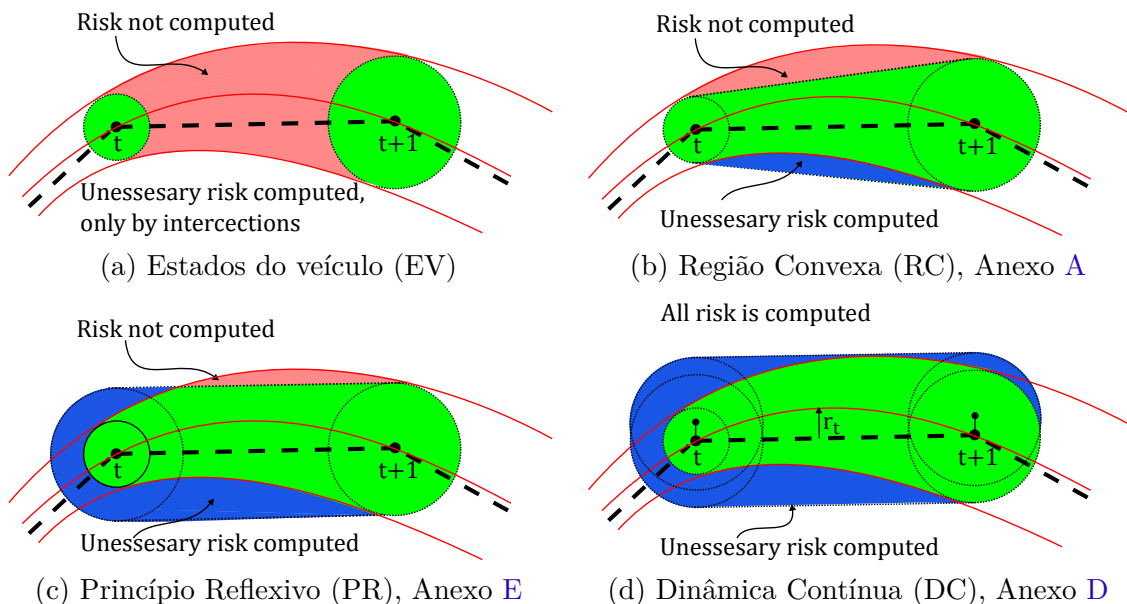


Figura 33 – Abordagens para tratar o desvio de obstáculos

Para validar cada proposta, foram realizados testes resolvendo os modelos PLIM em 50 mapas gerados aleatoriamente com  $J = 12$ ,  $T = 8$ ,  $\Delta t = 2.5s$  e nível de risco  $\Delta = 0.1$ .

Figura 34 apresenta exemplos de rotas encontradas por cada abordagem no mapa 06. Uma simulação de Monte Carlo é realizada para validar se o risco de colisão, de fato, está sendo respeitado, considerando um total de 10000 partículas (rotas). Nota-se que a primeira abordagem proposta por (BLACKMORE; ONO; WILLIAMS, 2011) falha e todas as demais são conservativas.

A Tabela 13 apresenta a média dos resultados para todos os 50 mapas. Considerando a simulação de Monte Carlo, a abordagem EV falha uma vez que a porcentagem de colisões foi de 74.11%, enquanto o risco máximo aceito foi especificado em  $\Delta = 0.1$ . Considerando o tempo médio para resolver o modelo, a abordagem RC é a mais demorada. O valor



Figura 34 – Exemplo de rotas diferentes encontradas em cada abordagem

da função objetivo para as rotas em RC, PR e DC são similares com uma pequena vitória de RC. Todos os modelos apresentam igual número de variáveis binárias (Bin), mas comparando a primeira proposta, RC, variáveis contínuas e restrições extras são adicionadas ao modelo. Tal acréscimo justifica a dificuldade em resolver este modelo. Por outro lado, a abordagem PR demanda menos restrições extras, tornando mais fácil a resolução do modelo ao levar em torno de 10.20 segundos em média. Por fim, DC consome um pouco mais de restrições, mas tem as garantias matemáticas de tratar corretamente o crescimento da incerteza, bem como a dinâmica contínua do veículo. Os modelos RC e PR são, normalmente, válidos e conservadores em média, como mostrado pelos valores da simulação de Monte Carlo. Porém, essas abordagens podem falhar em situações específicas visto que a matemática desenvolvida não contempla totalmente a dinâmica do veículo, existindo regiões de risco não computadas como ilustrado pela Figura 33.

Resumindo, o modelo DC apresenta garantias matemáticas maiores para lidar com o desvio de obstáculos, sendo resolvido em um tempo computacional intermediário. O modelo PR é rápido e, normalmente, válido para desvio de obstáculos, mas perde as garantias visto que utiliza como hipótese que a aeronave vai seguir linhas retas entre os pontos da rota. O modelo RC tem melhor custo de função objetivo e retorna rotas mais próximas ao valor ótimo, sendo normalmente válido no tratamento do desvio de obstáculos. Todavia, RC consome mais tempo computacional para ser resolvido e também perde as garantias por considerar hipóteses que podem não ser válidas para a dinâmica do veículo.

Tabela 13 – Resultados com valores médios para cada abordagem

Método	Monte Carlo	Tempo(s)	Objetivo	Bin	Cols	Rows
EV	74.11%	3.83	0.080	432	2664	1249
RC	1.12%	18.53	<b>0.308</b>	432	3048	2389
PR	0.31%	<b>10.20</b>	0.324	432	<b>2664</b>	<b>1633</b>
DC	<b>0.28</b> %	14.27	0.327	432	<b>2664</b>	2305

### 7.1.6 Abordagens com computação evolutiva

Esta seção apresenta a aplicação de técnicas de computação evolutiva para resolver o problema, um método chamado *Hybrid Genetic Algorithm* (HGA) foi desenvolvido e publicado no *Genetic and Evolutionary Computation Conference* (GECCO 2016), Anexo C. Esta abordagem foi comparada com métodos exatos e heurísticos. A solução exata é obtida ao resolver o PLIM do problema usando  $CNPP^{LEN}$ , onde é utilizado o algoritmo *branch and cut*, disponível no solver IBM ILOG CPLEX 12.2. A heurística é a Customized Solution Approach (CSA), proposta em (BLACKMORE; ONO; WILLIAMS, 2011).

Para o HGA, foi definido um tempo limite de execução de 10 segundos, enquanto para o  $CNPP^{LEN}$  e CSA foi utilizado um limite de uma hora, devido as limitações dessas abordagens em encontrar soluções. Por outro lado, o tempo reduzido de 10 segundos para o HGA foi estabelecido para avaliar se esse método é capaz de encontrar boas soluções com baixo tempo computacional. A ideia é possibilitar uma execução rápida o suficiente e capaz de ser embarcada posteriormente na arquitetura de hardware de um VANT. Os experimentos consideraram 50 mapas não regulares com  $J = 20$ ,  $T = 20$  e  $\Delta = 0.001$ .

Nas Figuras 35, são apresentadas duas rotas que o HGA determinou em duas execuções diferentes. Soluções diferentes podem ser retornadas pelo HGA, visto que é um método estocástico. Portanto, neste experimento o HGA foi executado 100 vezes para cada mapa.

O primeiro resultado compara o valor das funções objetivos das soluções do HGA,  $CNPP^{LEN}$  e CSA na Figura 36. Os valores médios das soluções do HGA são apresentados, em que também são apresentados os valores máximos e mínimos. As três abordagens têm resultados similares para a maioria dos mapas. Os piores resultados do HGA acontecem

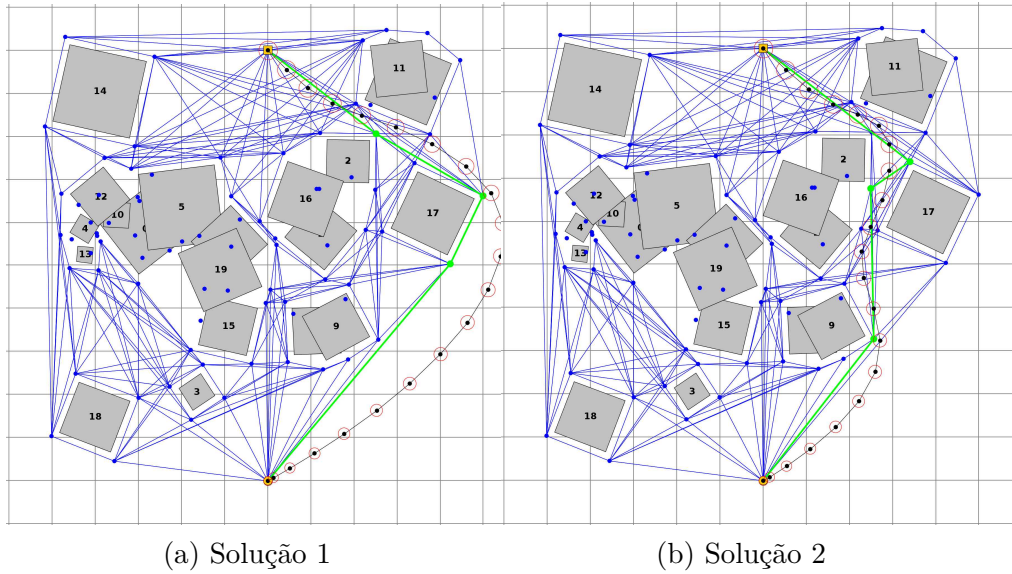


Figura 35 – Rotas encontradas em duas execuções diferentes do HGA.

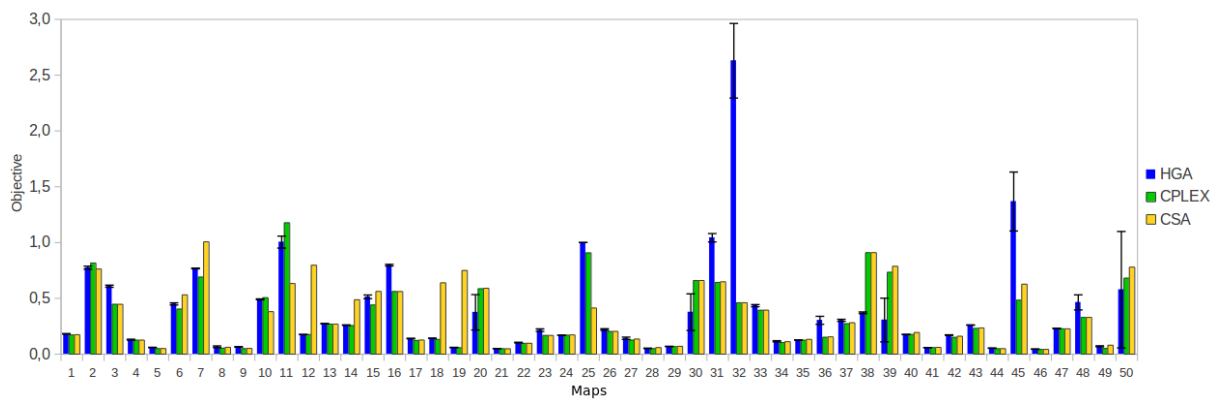


Figura 36 – Comparando o custo da função objetivo para todos os mapas.

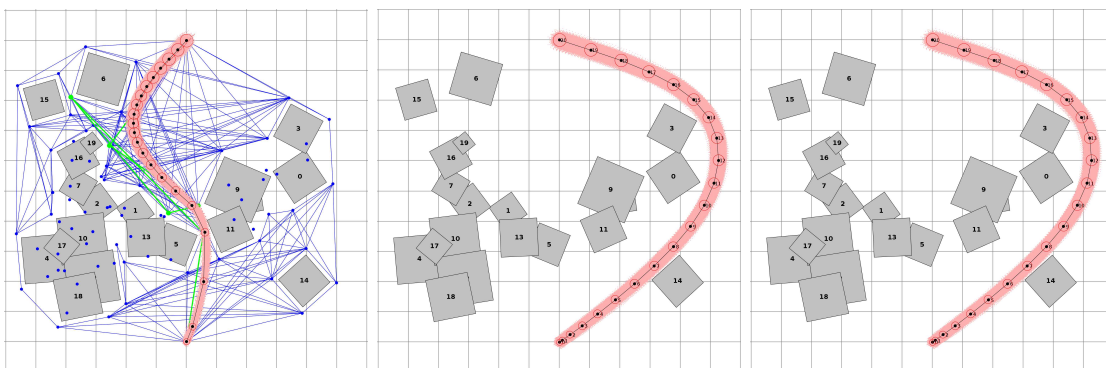


Figura 37 – Soluções encontradas pelo HGA, CPLEX e CSA no mapa 32.

nos mapas 32 e 45. A Figura 37 apresenta a pior solução do HGA no mapa 32 e as soluções obtidas pelo  $CNPP^{LEN}$  e CSA. O caminho do HGA é pior em termos de função objetivo, mas aparentemente não é uma solução inviável. O caminho do HGA passa entre os obstáculos 5 e 11, mas com as incertezas acumuladas ao longo da trajetória, foi necessário

aplicar aceleração máxima durante os primeiros instantes. Isso causa maior consumo de combustível e justifica o valor elevado da função objetivo do HGA. Apesar disso, nota-se que o HGA conseguiu uma solução dentro do tempo e respeita os riscos de colisão. Logo, tal rota é perfeitamente viável indicando que o HGA poderia ser adicionado como método para tomada de decisão dentro de uma plataforma embarcada.

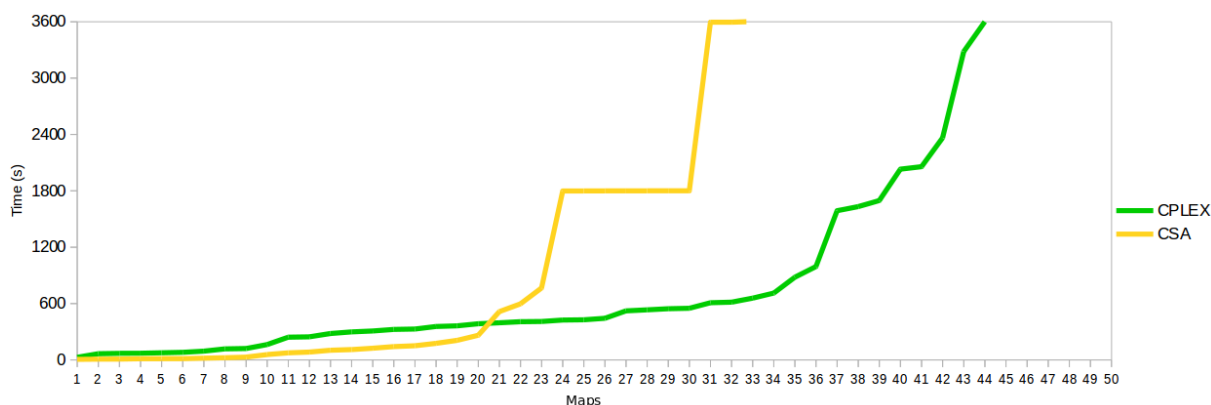


Figura 38 – Tempo gasto pelo o CPLEX e CSA para alcançar as soluções do HGA.

Figura 38 compara quanto tempo o  $CNPP^{LEN}$  e CSA levam para encontrar uma solução com custo igual ou melhor que as soluções do HGA. Os gráficos são organizados em ordem crescente de tempo, assim é possível perceber que o  $CNPP^{LEN}$  não foi capaz de superar o HGA em muitos mapas dentro de um limite de 1 hora. O CSA não superou o HGA dentro do limite de 1 hora para 17 mapas. O  $CNPP^{LEN}$  e CSA levam cerca de 600 segundos na maioria dos mapas para encontrar soluções melhores que o HGA. O  $CNPP^{LEN}$  gasta em média 1060 segundos e o CSA 1762 segundos para superar o HGA. Logo, o HGA se trata de um método mais promissor que as outras abordagens em aplicações que necessitem encontrar soluções rapidamente.

### 7.1.7 Resultados para processamento embarcado

Uma arquitetura embarcada é proposta como ilustrado na Figura 39. O sistema embarcado proposto tem 4 componentes de hardware: atuadores, piloto automático, sensores e processador embarcado. Os atuadores são responsáveis pelas manobras executadas pelo VANT, os sensores são responsáveis por capturar as informações do ambiente e promoverem a informação necessária a navegação e execução da missão. O piloto automático é responsável por fazer a navegação e manter a aeronave na rota programada. Por fim, um processador com dois núcleos de processamento é dedicado a execução de dois sistemas supervisores chamados de *Mission-Oriented Sensor Arrays* (MOSA) e *In-flight Awareness Systems* (IFA). O MOSA é responsável pelas decisões para cumprimento da missão e o IFA é responsável pela segurança do voo, tratamento de falhas e controle em situações adversas. O IFA pode assumir o controle total da aeronave, podendo abortar a missão ou mesmo,

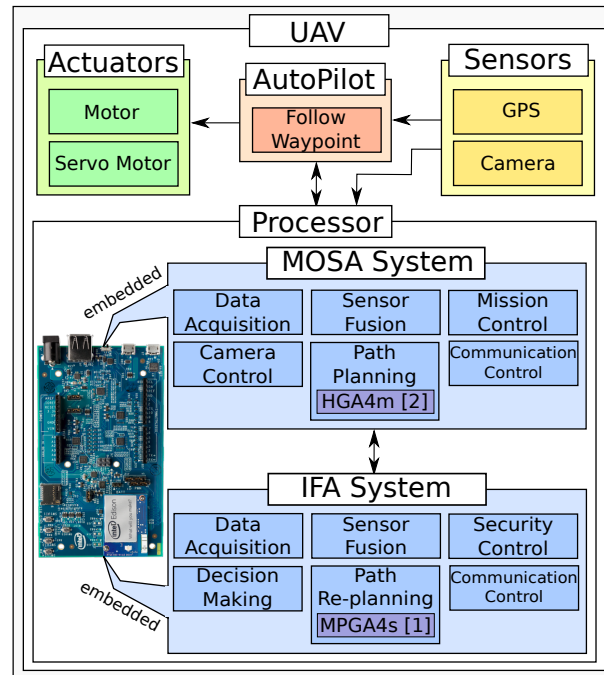


Figura 39 – Arquitetura embarcada para a aeronave.

definir um pouso de emergência. Um artigo descrevendo o sistema proposto foi aceito para apresentação e publicação no *Genetic and Evolutionary Computation Conference (GECCO 2017)*, Anexo H.

No módulo MOSA, foi incluído o algoritmo HGA apresentado na seção anterior que na publicação mencionada foi chamado de HGA4m. A performance do HGA4m foi avaliada quando executada em duas plataformas de hardware. A primeira delas é uma arquitetura embarcada considerando um processador Intel Edison dual core com 500 MHz, 1 GB RAM e sistema operacional Linux - Yocto. A segunda é uma arquitetura de computador pessoal (PC) tradicional que possui um processador Intel Core i5 com 1.8 GHz, 4 GB RAM e sistema operacional Linux - Ubuntu 16.04. Um total de 40 mapas foram, aleatoriamente, gerados considerando  $J = 20$ ,  $T = 20$  e  $\Delta = 0.001$ .

Desta vez os experimentos foram calibrados para serem coerentes com os limites de velocidade e aceleração do VANT Ararinha, descrito na seção 1.4. Os parâmetros relativos ao ararinha são apresentados na Tabela 14. Para os experimentos utilizando Software-In-The-Loop (SITL), integrado com o simulador FlightGear (FG), foi utilizada a aeronave Rascal 110, dada sua similaridade desta com o Ararinha, cuja dinâmica não está disponível para ser simulada no FG.

Os resultados para o HGA4m foram executados 10 vezes em cada mapa nas duas plataformas de hardware: Edison e PC. O método teve um tempo de execução limitado em 10 segundos. Duas métricas são comparadas: o número de soluções avaliadas e a distância percorrida. A Figura 40 ilustra a média do número de soluções avaliadas para cada mapa.



Tabela 14 – Principais parâmetros do Ararinha e Rascal.

Componente	Ararinha	Rascal 110
Envergadura	1.90m	2.79m
Comprimento	1.15m	1.92m
Potencia	740W	1600W
Peso	2.83kg	4.99kg
Payload	0.60kg	0.91kg
Autonomia	15 minutes	23 minutes

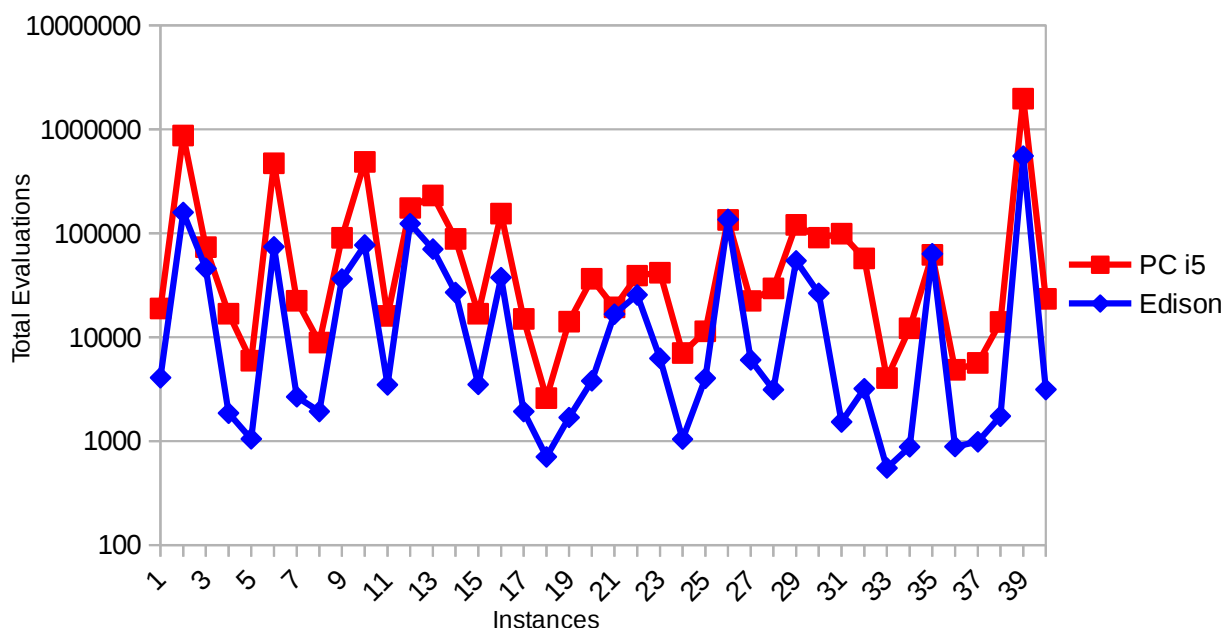


Figura 40 – Número de soluções avaliadas.

O número de soluções avaliadas é menor quando o HGA4m é executado na plataforma embarcada (Edison), como deveria ser esperado. Isso significa que HGA4m explorou menos o espaço de busca quando comparado ao seu uso em uma plataforma PC. Apesar disso, a qualidade das soluções em termos da segunda métrica, distância percorrida, é praticamente a mesma como ilustrado na Figura 41. A segunda métrica indica que a qualidade das rotas foi mantida mesmo quando o método é executado em uma plataforma de hardware inferior.

Tabela 15 – Kruskal-Wallis test for HGA4m

Métrica	Arquitetura	p-value
Avaliações	PC i5	< 0.0001
	Edison	
Distância	PC i5	0.185
	Edison	

O teste estatístico não paramétrico Kruskal-Wallis foi executado para avaliar os resultados das Figures 40-41, como apresentado na Tabela 15. Há uma diferença significativa no número de soluções avaliadas entre as plataformas PC e Edison, como pode

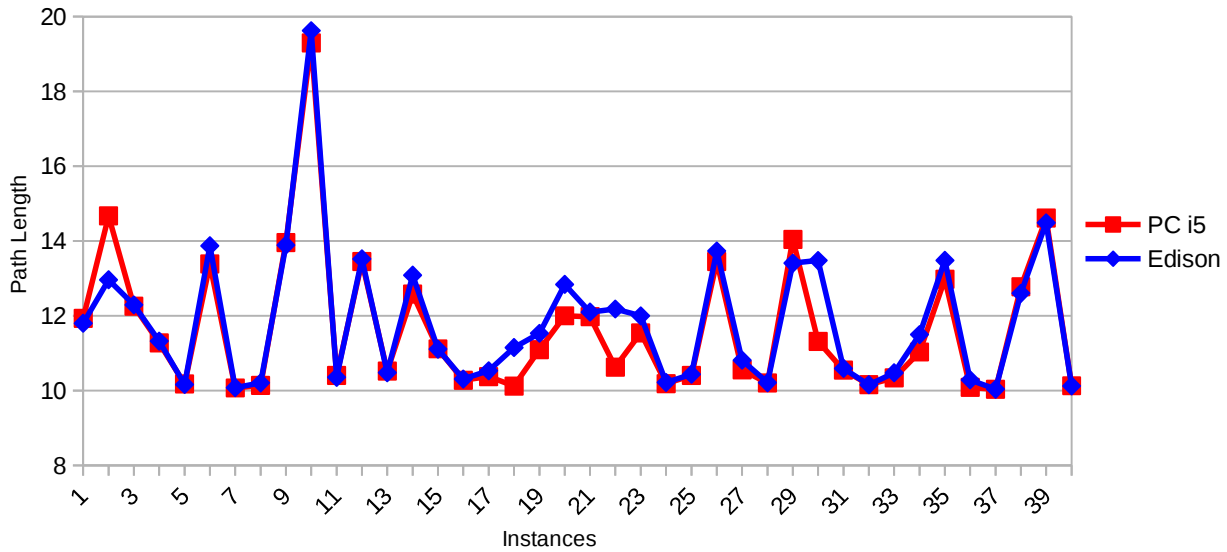


Figura 41 – Distância percorrida pelas soluções.

ser visto pelo  $p\text{-value} < 0.0001$ . Contudo, não foi encontrada uma diferença significativa em termos de distância percorrida (qualidade das soluções). Os critérios de independência, homoscedasticidade e normalidade precisam ser satisfeitos para aplicar testes estatísticos paramétricos. Se ao menos um critério não for satisfeito recomenda-se aplicar testes não paramétricos. O teste de Kolmogorov-Smirnov foi previamente executado e mostrou que os resultados apresentados nas Figuras 40 e 41 não seguem uma distribuição normal, o que justifica o uso do teste não paramétrico mencionado.

Agora são avaliados um estudo de caso em um cenário real, onde a arquitetura embarcada proposta foi avaliada com experimentos usando *Software-In-The-Loop* (SITL). A técnica de SITL é utilizada para validar a capacidade do piloto automático em seguir as trajetórias planejadas, onde o hardware do piloto automático é simulado. Este tipo de experimento permite verificar o comportamento da aeronave em condições bem próximas as encontradas durante um voo real. A Figura 42a ilustra o cenário da missão e a Figura 42b apresenta a rota executada. A rota completa executada pelo HGA4m é representada pelas linhas em verde. A análise da rota mostra que a missão foi executada sem violação ou colisão com obstáculos. Essa simulação com SITL pode ser vista no link<sup>2</sup>.

## 7.2 Resultados para o CCQSP

Os resultados dos testes apresentados nas seções a seguir foram executados utilizando um dos dois tipos de configuração de máquinas distintas descritas abaixo. Cada seção indicará qual configuração foi utilizada para manter a coerência na comparação com outros resultados da literatura.

<sup>2</sup> <<https://youtu.be/O-xMC51w5ec>>



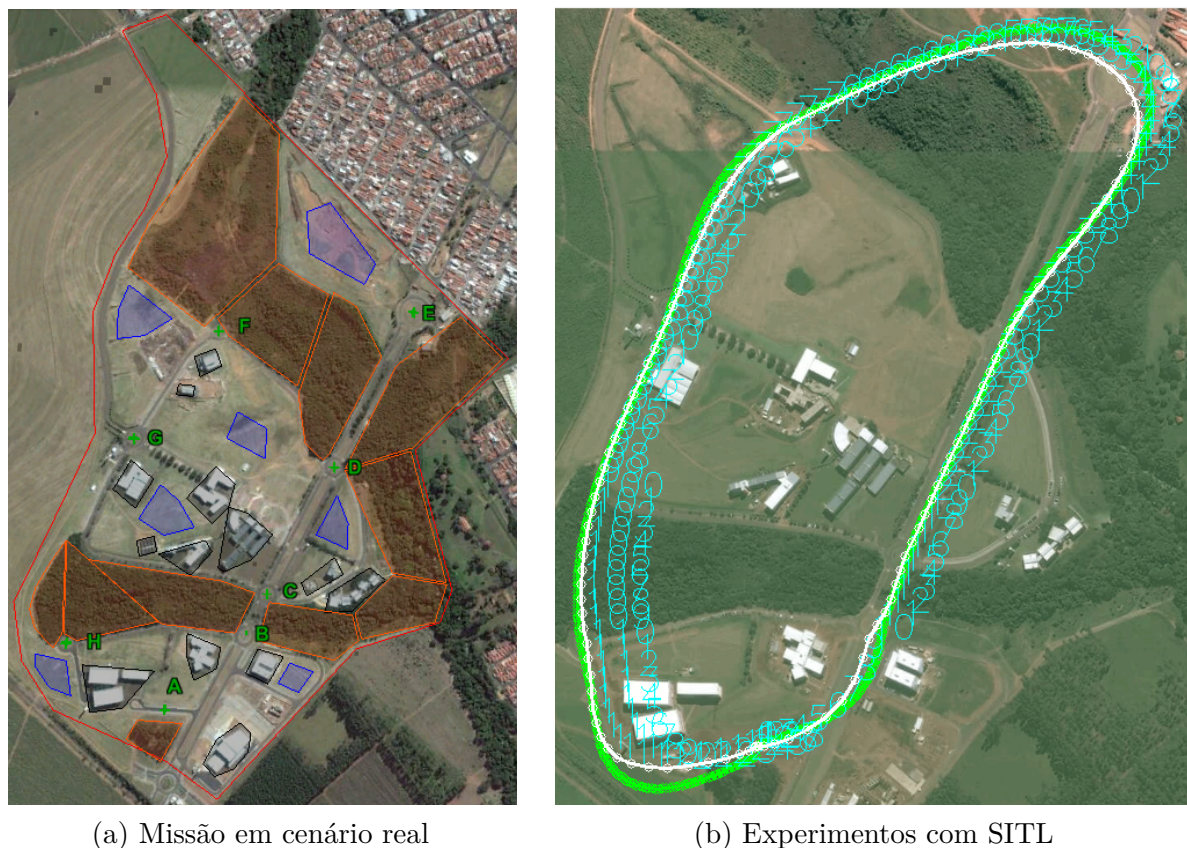


Figura 42 – Missão em cenário real e aplicação do SITL

**machine-1** Intel Core i7 CPU at 2.67 GHz and 8 GB of RAM with IBM ILOG CPLEX solver version 12.2. and run on Windows 7 OS.

**machine-2** Intel Core i7 CPU at 3.4 GHz and 32 GB of RAM with IBM ILOG CPLEX solver version 12.6. and run on Debian 8.2 OS.

### 7.2.1 Comparação do *pSulu* e propostas de *PLIM*

Os autores em (ONO; WILLIAMS; BLACKMORE, 2013) introduzem o CCQSP e propõem o algoritmo planejador *p-Sulu Planner*. Esse planejador utiliza *solvers* comerciais para solucionar as *chance-constrained* com um algoritmo *branch-and-bound* utilizado na resolução do problema de *scheduling* presente no CCQSP.

O tempo de convergência do *pSulu* reportado em (ONO; WILLIAMS; BLACKMORE, 2013) cresce exponencialmente com o aumento do número de obstáculos (Ⓞ) e pontos de passagem (Ⓜ). O planejador *pSulu* é avaliado a partir de um conjunto de 20 cenários iguais ao descrito na Figura 43, onde é possível perceber que o *pSulu* se mostra pouco escalável considerando o aumento de obstáculos e pontos de passagem. Os autores argumentam que resolver a escalabilidade do CCQSP é para um trabalho futuro.

A nossa proposta de solução do CCQSP foi transformá-lo de um modelo de

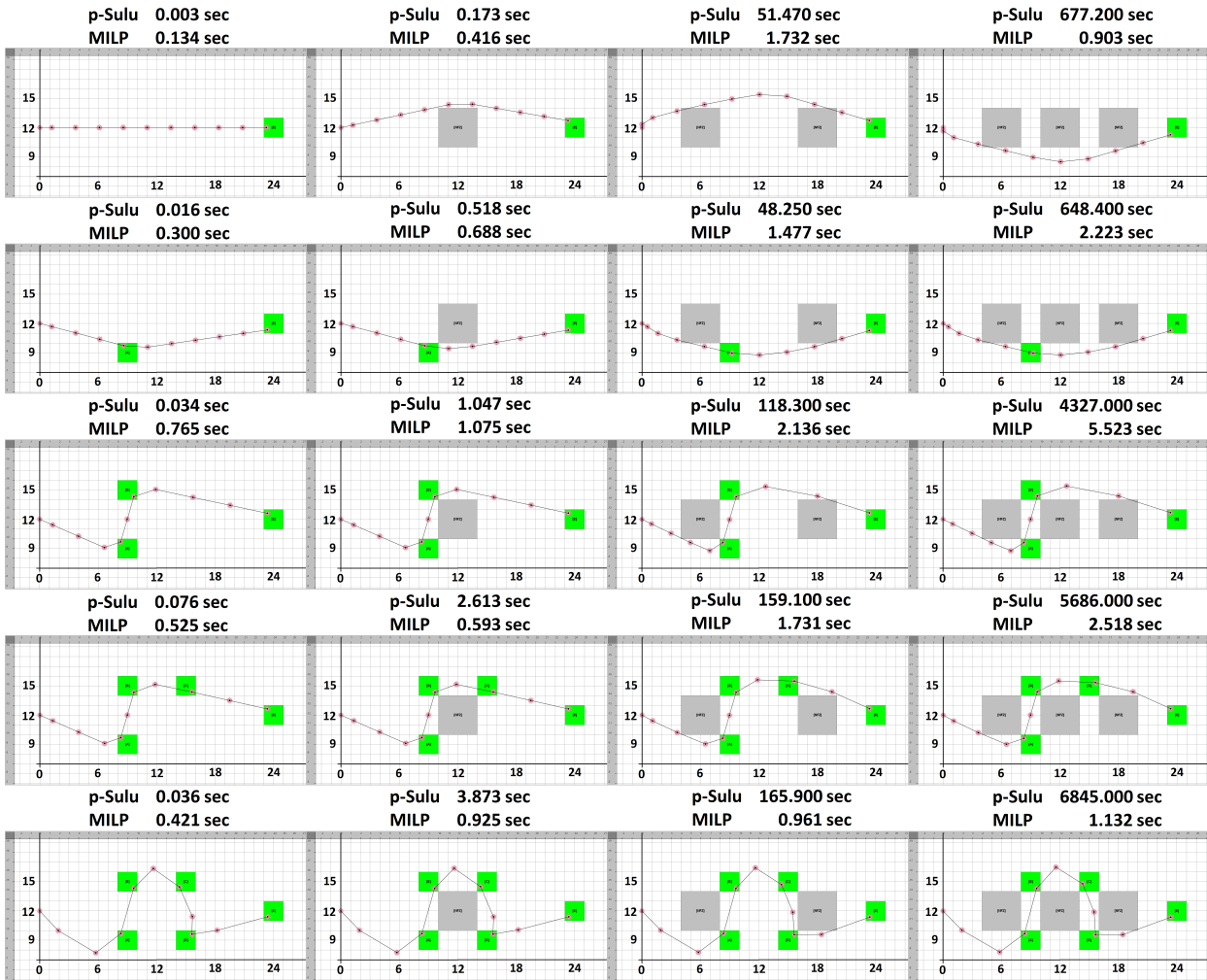


Figura 43 – pSulu and MILP comparison

Programação Não-Linear Inteiro Misto (PNLIM) para um modelo de Programação Linear Inteiro Misto (PLIM). A ideia é permitir que um método mais robusto de solução, como a *branch and cut*, fosse aplicado na determinação de controles aplicados e *scheduling*. O modelo PLIM proposto para o CCQSP foi descrito no capítulo 5 desta tese.

Para avaliar a força da abordagem aqui proposta, na mesma Figura 43, são apresentados os tempos de convergência em segundos com o uso do PLIM sobre a mesma configuração de máquina ( **machine-1** ). Nota-se uma superioridade relevante do PLIM em relação ao pSulu que chega a demorar quase duas horas para convergir (6845 sec), contra um pior caso de menos de seis segundos (5.523 sec) do PLIM .

### 7.2.2 Resultados das melhorias propostas

Nesta seção, será avaliado o efeito na convergência do modelo PLIM quando as melhorias propostas no capítulo 5 são aplicadas. Essas melhorias são brevemente descritas a seguir:

- +N trata-se da adição das novas restrições para colisão de obstáculos descritas em (ARANTES *et al.*, 2016a).
- +O é o uso da aproximação linear para a função objetivo quadrática descritas em (ARANTES *et al.*, 2016a).
- +D uso da dinâmica considerando a resistência do ar conforme descrito na seção 5.5.1.
- +C uso eficiente das *chance constraints* descrito na seção 5.5.2.
- +S fortalecimento dos episódios, apresentado na seção 5.5.3.
- +NO agrupamento de +N e +O, proposto em (ARANTES *et al.*, 2016a).
- +CS efeito obtido no tratamento conjunto de +C e +S.
- +All efeito obtido quando todas as melhorias são adicionadas ao modelo.

A Tabela 16 apresenta o tempo de convergência para cada melhoria nos 20 cenários propostos por (ONO; WILLIAMS; BLACKMORE, 2013). Todavia, desta vez, o número de time steps utilizado foi  $T = 20$ . Esse valor de  $T$  aumenta o nível de dificuldade para o PLIM resolver o cenário ao considerar um horizonte de planejamento melhor discretizado. A primeira coluna da tabela apresenta uma descrição para cada cenário ao indicar o número de pontos de passagem (I) e obstáculos (O). A segunda coluna apresenta o tempo em segundos, para a convergência do PLIM. As colunas de 3 até 7 indicam como fica o tempo de convergência com a adição de cada melhoria, separadamente. Os valores em negrito destacam os tempos mais rápidos para cada cenário. As linhas Bin, Cols e Rows apresentam o número médio de variáveis binárias, colunas e linhas na modelagem do problema.

Nota-se que a adição das restrições de fortalecimento +S reduzem o tempo computacional médio comparado aos demais fortalecimentos, quando aplicados isoladamente sobre o modelo. O número de variáveis binárias no modelo também é reduzido com +S, assim como o número de colunas e linhas em relação ao PLIM. A adição do uso eficiente das *chance constraints* com +C é o fortalecimento que reduz mais drasticamente as colunas e linhas do modelo. Por outro lado, a adição de +N e +O faz aumentar o número de linhas e colunas, respectivamente. Nas três últimas colunas da tabela estão os resultados combinando fortalecimentos, onde há uma redução mais acentuada no tempo de convergência. O método +NO representa a proposta dos autores publicada em (ARANTES *et al.*, 2016a). Nota-se que o modelo ficou com mais linhas e colunas. A nova proposta +CS combina a redução nas variáveis binárias junto com as linhas e colunas. Por fim, o modelo com todos os fortalecimentos atuando em conjunto +All (+N, +O, +D, +C, +S) retorna os melhores resultados em todos os cenários. Há uma melhoria considerável no tempo computacional com uma quantidade reduzida de variáveis binárias. Além disso, o número de colunas e linhas no modelo gerado se mantém em um valor razoável.

A Tabela 17 apresenta os resultados executados nas mesmas condições ( $T = 10$ ) que o método pSulu utilizando a configuração **machine-1**. Nas Tabelas 16 e 17, a linha Avg apresentava a média dos tempos considerando os 20 cenários. Percebemos uma variação grande nos tempos entre os diferentes cenários.

Tabela 16 – Impacto de cada melhoria no tempo de execução para cada cenário

Mapas	PLIM	+N	+O	+D	+C	+S	+NO	+CS	+All
I1O0	0.34	0.36	0.41	0.30	<b>0.17</b>	0.35	0.40	<b>0.19</b>	0.29
I2O0	0.85	0.85	0.71	1.26	<b>0.51</b>	0.74	0.72	0.48	<b>0.37</b>
I3O0	5.64	5.44	8.11	4.24	3.86	<b>1.96</b>	8.83	1.00	<b>0.80</b>
I4O0	6.62	6.31	4.81	5.34	<b>2.64</b>	3.17	5.00	<b>1.57</b>	<b>1.57</b>
I5O0	10.25	10.08	8.80	<b>3.74</b>	7.95	3.80	9.39	2.66	<b>1.87</b>
I1O1	1.23	2.13	1.36	5.38	<b>0.55</b>	1.40	2.61	<b>0.95</b>	1.04
I2O1	<b>2.81</b>	4.40	4.33	23.57	5.41	2.94	5.24	1.20	<b>0.86</b>
I3O1	43.86	74.67	18.77	35.38	23.25	<b>14.43</b>	20.70	4.02	<b>3.71</b>
I4O1	23.85	49.46	13.32	34.67	21.11	<b>8.85</b>	38.05	8.89	<b>4.61</b>
I5O1	52.32	14.30	39.32	<b>13.99</b>	30.51	22.91	18.50	6.48	<b>3.53</b>
I1O2	32.55	78.99	5.38	21.78	18.14	<b>5.33</b>	10.92	6.34	<b>2.19</b>
I2O2	27.76	69.37	<b>4.34</b>	67.33	10.86	43.63	16.70	7.31	<b>2.74</b>
I3O2	<b>44.26</b>	98.61	69.59	50.70	59.42	70.86	31.57	20.14	<b>7.10</b>
I4O2	76.04	60.73	62.75	126.43	<b>32.84</b>	100.90	65.62	33.19	<b>6.46</b>
I5O2	82.17	22.31	74.21	35.96	59.25	<b>21.73</b>	28.82	31.05	<b>4.09</b>
I1O3	11.87	60.57	<b>7.78</b>	80.11	14.83	86.79	15.53	18.98	<b>2.88</b>
I2O3	30.13	55.79	<b>10.28</b>	82.82	17.65	50.37	64.02	4.44	<b>4.38</b>
I3O3	189.54	133.23	<b>67.94</b>	143.78	216.66	123.39	39.21	18.49	<b>8.56</b>
I4O3	129.68	167.88	115.33	243.56	111.96	<b>40.24</b>	88.15	29.47	<b>9.20</b>
I5O3	241.34	<b>27.14</b>	246.18	46.94	138.13	41.32	40.41	28.67	<b>3.69</b>
Avg	50.66	47.13	38.19	51.36	38.79	<b>32.26</b>	25.52	11.28	<b>3.50</b>
Bin	409	409	409	409	409	<b>315</b>	409	<b>315</b>	<b>315</b>
Cols	11034	11154	12394	11034	<b>1669</b>	11003	12514	<b>1637</b>	3102
Rows	3845	4235	3965	3845	<b>1937</b>	3769	4323	<b>1861</b>	2339

Tabela 17 – Avaliando pSulu e melhorias propostas.

Maps	pSulu	PLIM	+NO	+CS	+All
I1O0	<b>0.003</b>	0.134	0.163	0.088	0.138
I2O0	<b>0.016</b>	0.300	0.278	0.159	0.192
I3O0	<b>0.034</b>	0.765	0.605	0.198	0.261
I4O0	<b>0.076</b>	0.525	0.780	0.281	0.295
I5O0	<b>0.036</b>	0.421	0.637	0.224	0.344
I1O1	<b>0.173</b>	0.416	0.451	0.288	0.298
I2O1	0.518	0.688	0.636	<b>0.323</b>	0.399
I3O1	1.047	1.075	1.422	<b>0.426</b>	0.500
I4O1	2.613	0.593	1.161	<b>0.331</b>	0.607
I5O1	3.873	0.925	0.915	<b>0.400</b>	0.529
I1O2	51.470	1.732	1.105	0.508	<b>0.421</b>
I2O2	48.250	1.477	1.028	0.400	<b>0.385</b>
I3O2	118.300	2.136	1.187	1.640	<b>0.642</b>
I4O2	159.100	1.731	1.311	<b>0.574</b>	0.896
I5O2	165.900	0.961	0.769	0.478	<b>0.428</b>
I1O3	677.200	0.903	0.751	1.787	<b>0.417</b>
I2O3	648.400	2.223	0.808	<b>0.698</b>	0.745
I3O3	4327.000	5.523	1.316	1.851	<b>1.084</b>
I4O3	5686.000	2.518	1.893	<b>0.615</b>	1.314
I5O3	6845.000	1.132	0.768	0.809	<b>0.463</b>
Avg	936.750	1.309	0.899	0.604	<b>0.518</b>
Worst	6845.000	5.523	1.893	1.851	<b>1.314</b>
Bin	-	214	214	<b>165</b>	<b>165</b>
Cols	-	5667	6407	<b>850</b>	1582
Rows	-	1982	2221	<b>966</b>	1204



Apenas a média não é suficiente estatisticamente para afirmar se houve, de fato, mudança no tempo de convergência. Para aplicar testes estatísticos paramétricos, os critérios de independência, homoscedasticidade e normalidade precisam ser satisfeitos pelas amostras. As amostras são independentes se a ocorrência de uma não afeta a probabilidade de outra. Esse critério pode ser verificado pelo coeficiente de correlação de Pearson. A homoscedasticidade ocorre se os grupos de amostras possuem a mesma variância, podendo ser verificado pelo teste de Levene. A normalidade é satisfeita se a população da qual a amostra foi obtida segue uma distribuição normal. O método de Anderson Darling pode ser aplicado para verificar isso.

Se ao menos um critério não for satisfeito, não é possível garantir os resultados caso seja aplicado testes estatísticos paramétricos. Nesse caso, uma alternativa passa a ser o uso de testes não paramétricos. Nos experimentos, aqui realizados, todos os critérios falharam. Por isso, foi utilizado o teste não paramétrico de Friedman para determinar se há diferenças significativas nos resultados. O teste de Friedman determina se ao menos um método é significativamente diferente dos demais através de um P-Value menor que 0.05. Assim, rejeita-se a hipótese nula e a hipótese alternativa é que ao menos um método vem de uma população diferente. Um valor de P-Value = 0.0001 < 0.05 foi obtido a partir dos resultados obtidos, indicando que há diferença significativa entre os métodos apresentados nas Tabelas 17 e 16.

O teste de Nemenyi (DEMSAR, 2006) pode ser aplicado para comparações pareadas, indicando se há diferença significativa entre os métodos. O resultado obtido ao se aplicar esse teste é apresentado na Tabela 18. Os fortalecimentos +CS e +All são melhores que pSulu, PLIM e +NO. Não há diferença significativa entre +CS e +All.

Tabela 18 – Teste de Nemenyi para as Tabelas 17 e 16.

method	N	From Table 17		From Table 16	
		Sum of Ranks	Groups	Sum of Ranks	Groups
pSulu	20	72.0	B	-	-
PLIM	20	79.0	B	72.0	B
+NO	20	70.0	B	65.0	B
+CS	20	38.0	A	40.5	A
+All	20	41.0	A	22.5	A

Vale ressaltar que as melhorias no modelo não necessariamente facilitarão a resolução do problema, podendo apresentar em muitos casos um impacto negativo no tempo de convergência. Deve-se também considerar que as rotas encontradas podem ser diferentes. Por exemplo, o uso de +D busca incorporar ao modelo uma dinâmica mais realista para refletir uma melhor tomada de decisão, logo a rota ótima nesse contexto pode ser outra. As únicas mudanças que não afetam uma equivalência com as soluções retornadas pelo modelo PLIM são: +C, +S e obviamente +CS. A Figura 44 exemplifica isto ao mostrar as diferentes rotas retornadas para o cenário I303.

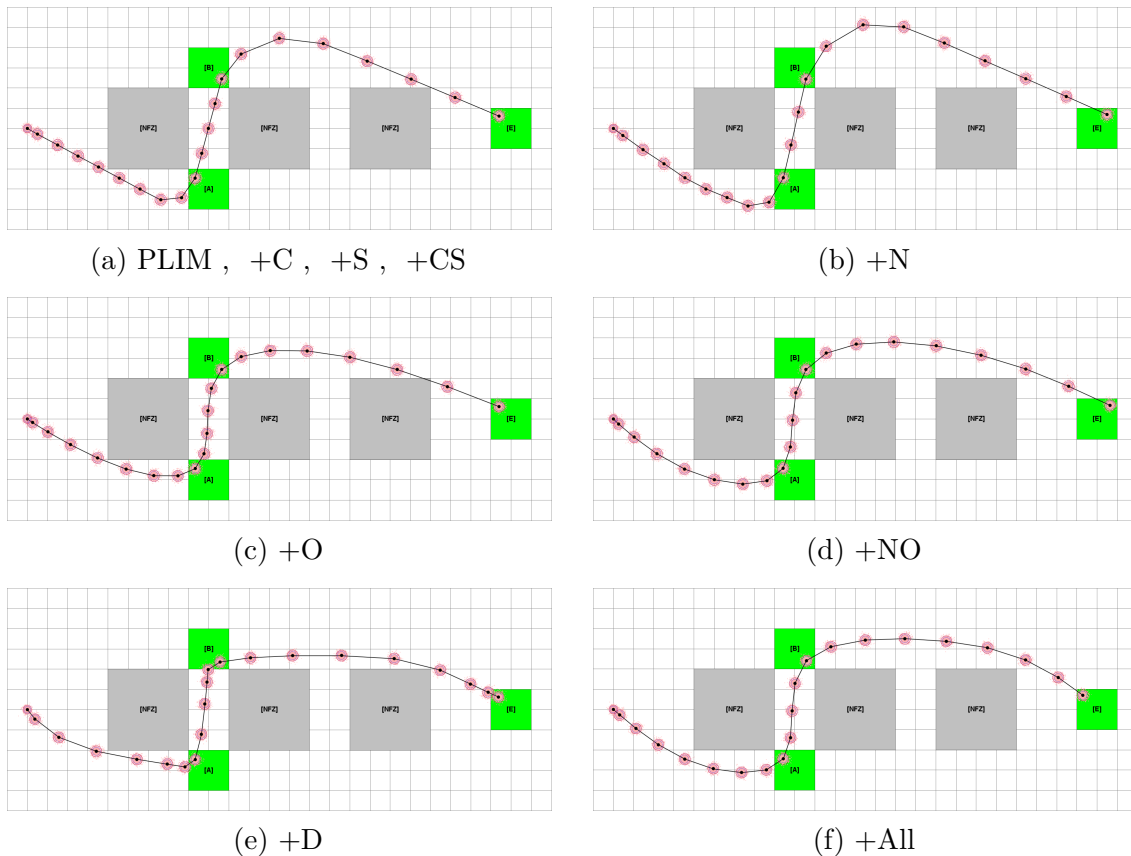


Figura 44 – Diferentes rotas para cada mudança no modelo

Algumas melhorias no modelo podem ser consideradas obrigatórias para permitir a obtenção de soluções viáveis. Por exemplo, a inclusão de +N proposta em (ARANTES *et al.*, 2016a) busca resolver o problema da colisão com os cantos dos obstáculos. Isso pode ser visualizado em todas as figuras nas quais esta melhoria não foi aplicada: 44a, 44c e 44e. A melhoria +O busca a execução de curvas mais suaves, pois penaliza acelerações bruscas do VANT. A melhoria +D evita que o efeito da inércia atue livremente sem penalidade ao voo.

### 7.2.3 Resultados de escalabilidade

Avaliando os métodos pSulu, PLIM, +NO, +CS e +All, percebe-se que apenas as propostas +NO definidas em (ARANTES *et al.*, 2016a) e +All são coerentes o suficiente para serem utilizadas como soluções válidas para o problema. Elas são as únicas que estabelecem consistentemente o desvio de obstáculos livre de colisões. Assim, a capacidade e escalabilidade da nova proposta +All será comparada com a escalabilidade de +NO definida em (ARANTES *et al.*, 2016a). Para isso, um total de 50 cenários diferentes foi definido para cada uma das situações descritas a seguir.

A escalabilidade das propostas +All e +NO será avaliada considerando aumentos

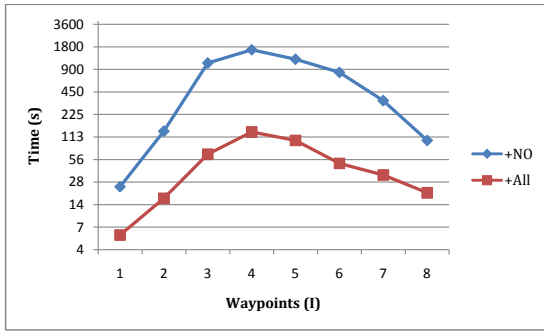
no número de waypoints  $\mathbb{I}$ , obstáculos  $\mathbb{O}$ , hiperplanos  $H$  e discretização do horizonte de planejamento (*time steps* ou *waypoints*)  $T$ . Para cada caso, foram validadas 8 variações nestes parâmetros, como apresentado na Figura 45, enquanto eram mantidos constantes os demais parâmetros. Os valores considerados foram  $\mathbb{O} = 3$ ,  $\mathbb{I} = 3$ ,  $H = 4$  e  $T = 16$ , levando a um total de 1600 cenários diferentes.

Figura 45 apresenta o resultado médio de escalabilidade considerando o tempo computacional e o gap em relação ao limitante inferior para as propostas +NO e +All. O tempo total de execução foi limitado em 1 hora. Os resultados de tempo estabelecem o valor médio de tempo para retornar soluções ótimas (gap = 0) ou a melhor solução factível dentro do limite de tempo. O gap é um valor em porcentagem e indica em termos de resultados médios uma prova para a otimalidade das soluções.

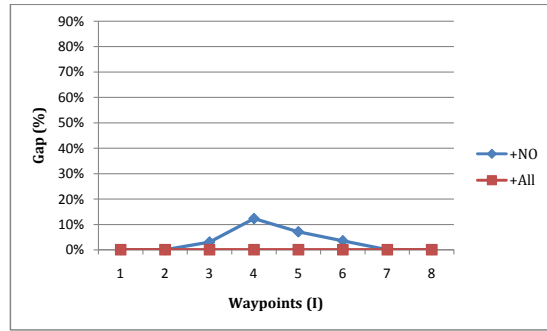
As duas propostas de melhoria se mostram escalável considerando um aumento no número de waypoints  $\mathbb{I}$  como ilustrado nas Figuras 45 (a)-(b), onde +All se destacou por conseguir encontrar o ótimo em todos os casos e possuir menor tempo médio de execução. +All retorna soluções abaixo de 113 segundos contra 1800 segundos de +NO no pior caso. Considerando o número de hiperplanos  $H$ , Figuras 45 (c)-(d), há uma dificuldade de +NO conseguir provar a otimalidade, levando o modelo a gastar em média todo tempo de execução. Nesse caso, o +All se manteve com resultados ótimos e os tempos de execução apresentaram um crescimento proporcional ao aumento de  $H$ .

Os problemas maiores aparecem ao escalar o número de obstáculos  $\mathbb{O}$  e *time steps*  $T$  como pode ser visto nas Figuras 45 (e)-(h), onde as duas melhorias propostas encontram maior dificuldade. Ao aumentar o número de obstáculos e *time steps*, +NO e +All apresentam um crescimento rápido no tempo de execução, onde a curva de crescimento do +All permaneceu abaixo da curva de +NO. Elas se aproximam no limite estabelecido de 3600 segundos. Os resultados de otimalidade indicam que o +All é superior ao +NO, conseguindo uma melhor escalabilidade e encontrando assim soluções mais próximas do valor ótimo.

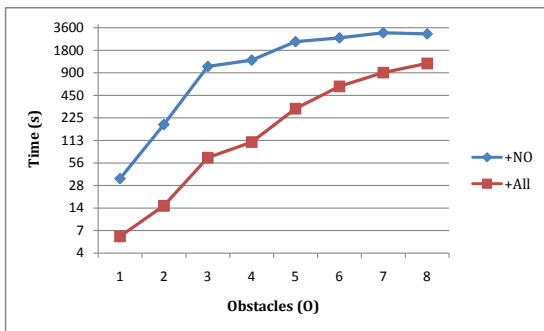
Para ilustrar melhor as causas das diferenças entre esses modelos, a Figura 46 apresenta o impacto da escalabilidade dos cenários no número de variáveis binárias, colunas e linhas dos modelos gerados por +NO e +All. O modelo +All é mais enxuto, começando com uma quantidade menor de variáveis, colunas e restrições. Por outro lado, o aumento em  $\mathbb{I}$ ,  $\mathbb{O}$ ,  $H$  e  $T$  sempre leva a um crescimento ainda maior do modelo gerado por +NO. Logo, +NO será maior que +All em termos de tamanho do problema e assim é esperado que +NO seja mais difícil de resolver do que +All.



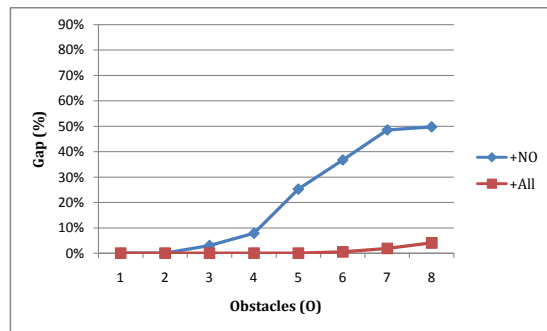
(a)



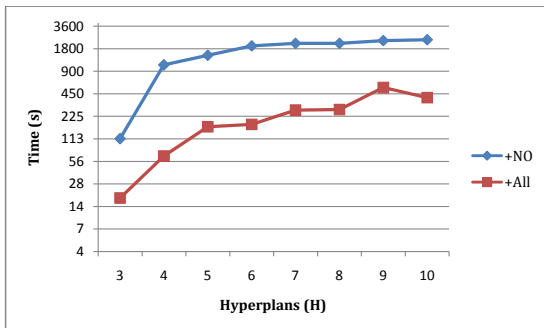
(b)



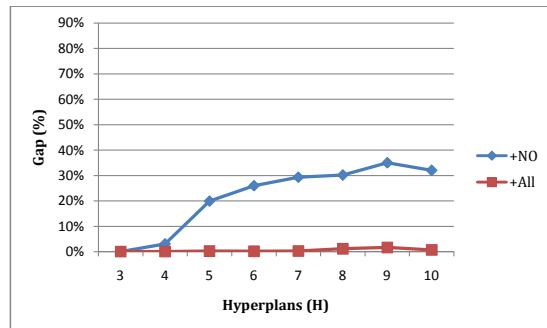
(c)



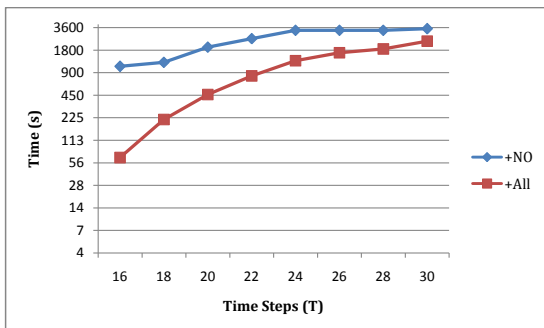
(d)



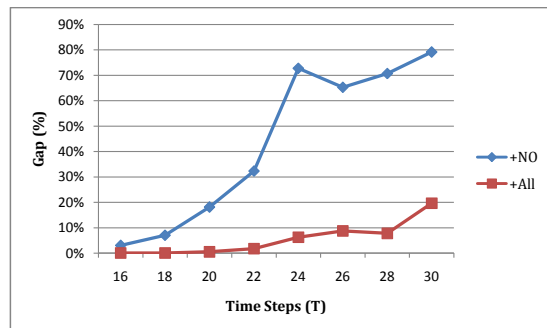
(e)



(f)



(g)



(h)

Figura 45 – Análise de escalabilidade para  $\mathbb{I}$ ,  $\mathbb{O}$ ,  $H$  e  $T$



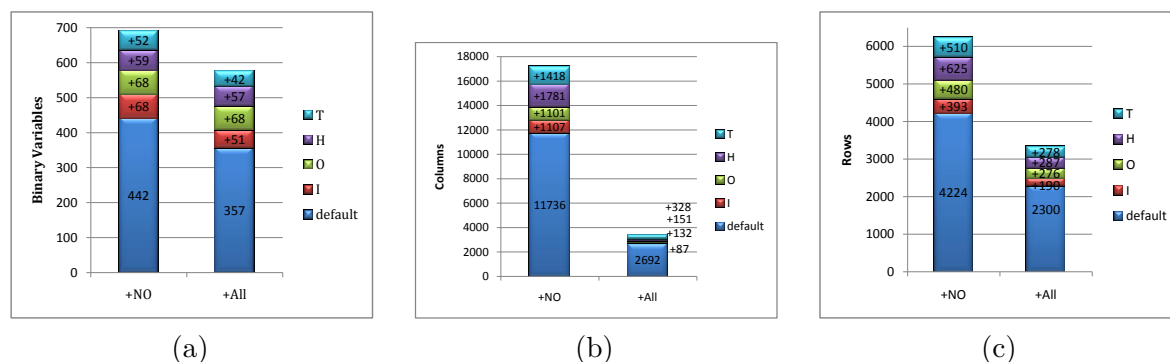


Figura 46 – Número de variáveis binárias, linhas e colunas adicionadas no modelo a cada passo.

## 7.3 Resultados para o HCCQSP

Os resultados do HCCQSP considerando nesta seção são apresentados para o cenário proposto por (LI, 2010) e descrito na seção 6.1. A Figura 47 estabelece a resolução do PLIM, considerando apenas a transformação do problema de um PNLIM para um PLIM, como estabelecido na seção 6.3. Para conseguir cumprir a missão dentro de uma discretização satisfatória, foi definido  $T = 60$  e  $\Delta t = 1s$ , já que o cumprimento de toda a missão exige a execução de várias idas e vindas entre as etapas a serem realizadas. Logo, valores baixos de  $T$  para missões com esta complexidade se mostram ineficazes. Todos os demais parâmetros da missão foram especificados conforme consta na descrição original do problema proposto em (LI, 2010) e detalhados na seção 6.2.

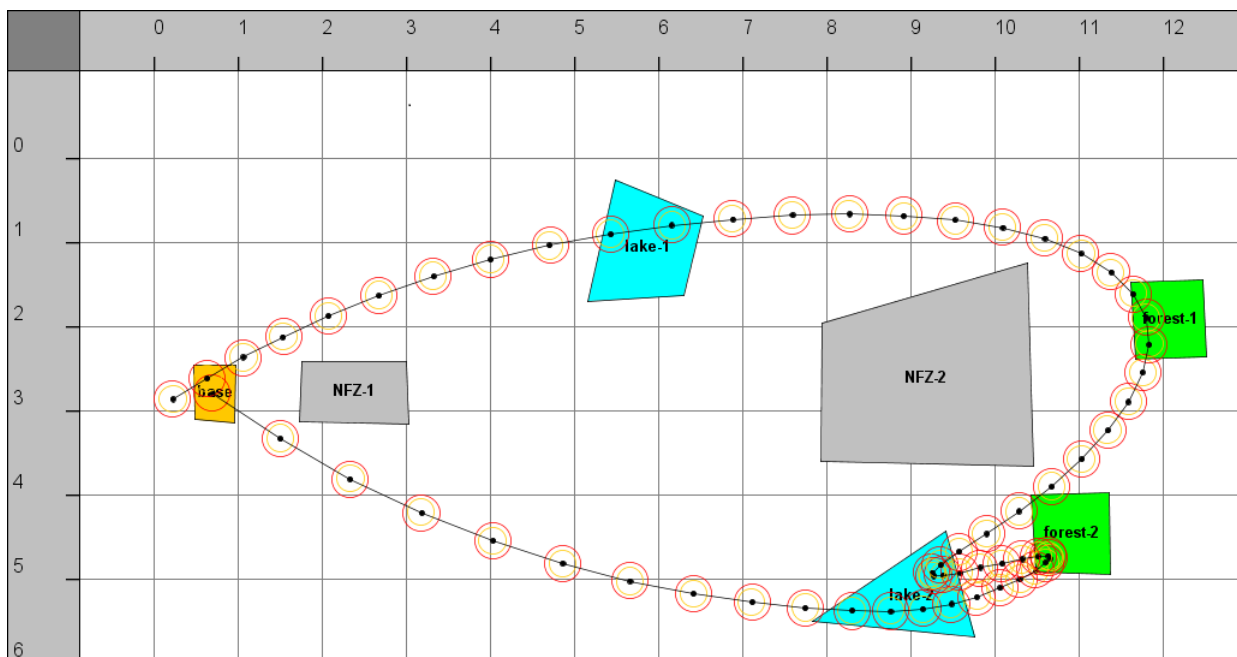


Figura 47 – Solução para HCCQSP, PLIM sem melhorias (factível em 7200.0 seg, gap 100%)

A solução apresentada na Figura 47 foi encontrada aos 6839 segundos, onde o solver CPLEX foi executado com um limite de duas horas (7200 s) estabelecido, sobre a mesma

configuração de máquina **machine-1** apresentada para o CCQSP na seção 7.2. Após este tempo de execução, a resolução do modelo PLIM parou sem a prova da otimalidade da solução encontrada aos 6839 segundos. Um gap de 100% foi retornado dado que não se conseguiu melhoria em relação ao limitante inferior ( $lowerbound = 0.0$ ).

O próximo passo foi a avaliação do comportamento do modelo PLIM ao se aplicar as melhorias para redução da complexidade descritas na seção 5.6. O resultado é apresentado na Figura 48. A solução ótima para o problema passou a ser encontrada, provando-se sua otimalidade em um tempo reduzido para 15.7 segundos. A principal explicação para a melhoria relevante obtida está na análise de complexidade apresentada na seção 5.6. A análise de complexidade descreve uma garantia matemática para redução da complexidade, reduzindo as variáveis binárias, contínuas e restrições do PLIM ao redefinir algumas variáveis e restrições do modelo. Assim, no PLIM da Figura 47, há 2668 variáveis binárias, 11737 variáveis contínuas e 24124 restrições, enquanto o PLIM da Figura 48 apresenta 858 variáveis binárias, 7007 contínuas e 9194 restrições.

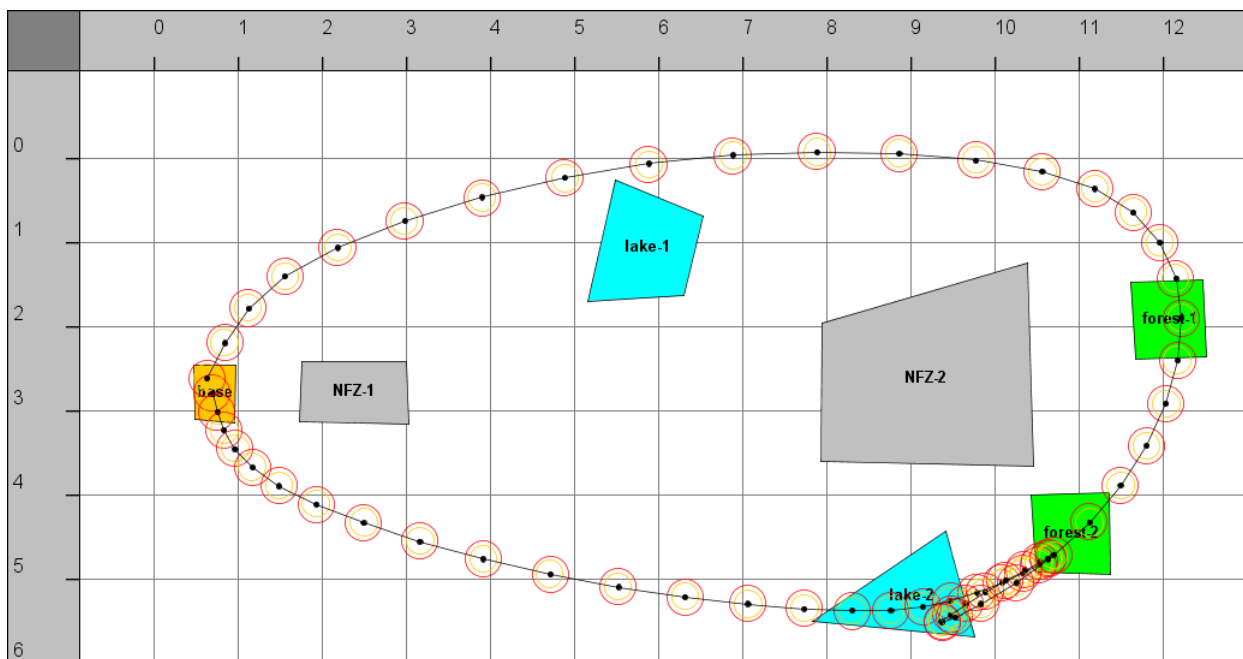


Figura 48 – Solução para HCCQSP, PLIM com melhorias (ótimo em 15.7 seg)

A Figura 49 apresenta os resultados de escalabilidade do HCCQSP considerando o aumento de Obstáculos (⊙), Pontos de Passagem (Ⓜ) e Time Steps ( $T$ ). Nota-se que é possível solucionar o problema com até  $T = 240$  praticamente sem impacto no tempo de execução. O aumento no número de obstáculos também teve o crescimento amortizado. Esses avanços se deram principalmente pela introdução no HCCQSP das equações apresentadas na seção 5.6.

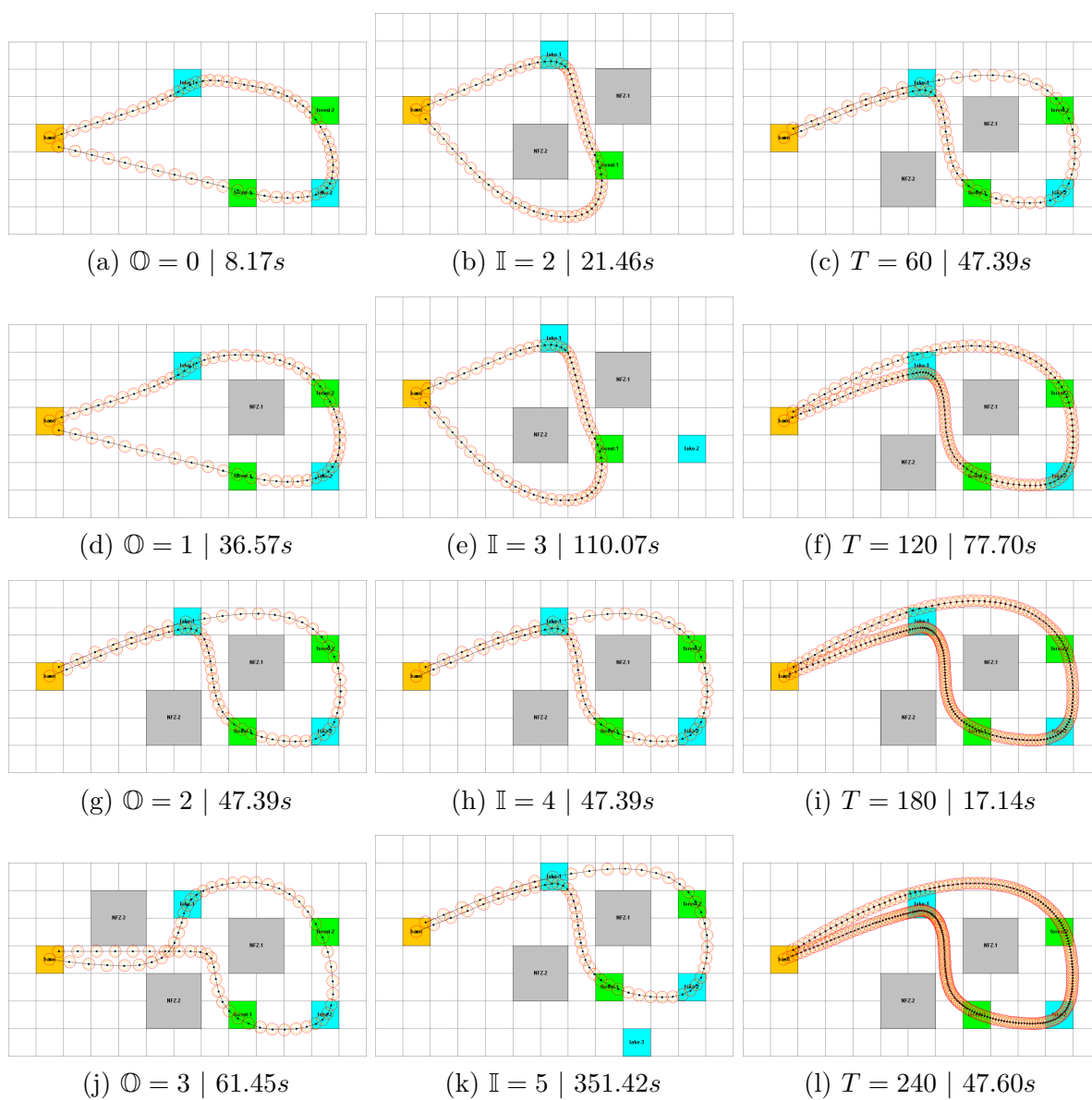


Figura 49 – Escalabilidade para o HCCQSP



---

## CONCLUSÃO

---

Os avanços alcançados ao solucionar o *Hybrid Qualitative State Plan (HQSP) Problem* foram descritos nesta tese. Tais avanços foram obtidos através da aplicação de técnicas baseadas em modelagem matemática, em que a aplicação de lógicas bem formuladas para o problema possibilitou que modelos complexos fossem representados e implementados de forma mais intuitiva. Algumas soluções encontradas foram validadas em simuladores de voo e embarcadas em hardware compatível com as características de um VANT de asa fixa, o Ararinha.

Trabalhos da literatura têm utilizado Programação Linear Inteira-Mista (PLIM) apenas em parte dos modelos CCNP, CCQSP e HQSP, visando a construção de heurísticas ou a aplicação do algoritmo *branch-and-bound* para uma solução exata. Isso acontece já que o problema é não linear e não convexo, assim o *branch-and-bound* tem sido utilizado para resolver a não convexidade e *solvers* não lineares são aplicados para as restrições não-lineares. Esta tese é pioneira ao apresentar um modelo completo de PLIM para o problema, viabilizando, assim, o uso do algoritmo *branch-and-cut*.

A não convexidade é resolvida com a adição de variáveis binárias artificiais e a não linearidade é aproximada usando função linear por partes. Esses avanços transformaram os modelos de Programação Não-Linear Inteira-Mista (PNLIM) em modelos de PLIM. Um PLIM é uma modelagem mais forte que um PNLIM, pois permite que técnicas eficazes sejam utilizadas para a resolução do problema. Tal abordagem permitiu avançar em termos de representatividade e escalabilidade. As questões relacionadas à representação e escalabilidade foram tratadas nesta tese a partir dos modelos PLIM estabelecidos.

A representação foi aprimorada ao se adicionar restrições ao modelo que permitiram tratar a alocação de risco de forma mais precisa. Inicialmente, o modelo proposto em (BLACKMORE; ONO; WILLIAMS, 2011) permitia o “corte” nos cantos dos obstáculos. Isso revelava que a alocação de risco, mesmo quando o “corte” não ocorria, poderia estar

sendo subestimada. Os resultados computacionais revelaram que o risco total das rotas era subestimado, passando a ser conservativo com a introdução das melhorias adicionadas à representação do problema nos três modelos PLIM propostos. A representação aqui adotada também passou a considerar mapas 3D, ou seja, com diferentes alturas. Por isso, a presente tese colaborou para se atingir o objetivo de melhorar a representação, ao permitir encontrar soluções viáveis de serem aplicadas ao mundo real.

A tese também atingiu os objetivos propostos em termos de escalabilidade. Isso se comprova pelo fato de soluções viáveis para cenários mais complexos terem sido obtidas dentro de um tempo computacional razoável. Os modelos PLIM e as melhorias propostas permitiram solucionar cenários com um maior número de obstáculos e horizonte de planejamento. Além disso, as heurísticas e meta-heurísticas introduzidas conseguiram solucionar tais problemas a partir de sua combinação com a resolução de modelos de programação linear relacionados. Alguns desses métodos poderiam inclusive ser embarcados em um VANT, dada a rapidez na obtenção das soluções como pode ser atestado pelas simulações realizadas.

Trabalhos futuros podem avançar em melhores resultados de tempo e escalabilidade com uso de métodos heurísticos, metaheurísticos e programação matemática, explorando características aqui não avaliadas. Otimização por enxame de partículas tem sido aplicada com sucesso em diversos trabalhos que envolvem solução de problemas em grafo. Métodos de programação matemática baseados em geração de colunas podem tirar proveito de restrições problemáticas como as restrições de *scheduling* e restrições que envolvem as ações híbridas.

Outra linha de trabalhos futuros está em explorar melhor as características inerentes a parâmetros criados para a tese, durante as transformações e otimizações realizadas. Por exemplo, para utilizar PLIM, muitas restrições utilizam o conhecido Big M para serem modeladas. Há trabalhos que propõem soluções alternativas e mais eficazes em oposição ao uso de Big M. Outro parâmetro que pode ser melhor ajustado é o  $\hat{S}$  descrito na seção 5.6. Esse parâmetro é responsável pela redução da complexidade do problema em termos de número de obstáculo. Uma escolha otimizada deste parâmetro pode reduzir ainda mais a complexidade na resolução de cenários com muitos obstáculos.

Todos códigos e cenários utilizados nesta tese estão disponíveis para download em <https://github.com/marcio-da-silva-arantes/PhDThesis>

## REFERÊNCIAS

---

---

ALTEROVITZ, R.; SIMEON, T. T.; GOLDBERG, K. The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty. In **Proc. Robotics: Science & Systems**, 2007. Citado nas páginas 28 e 29.

AOUDE, G. S.; LUDERS, B. D.; JOSEPH, J. M.; ROY, N.; HOW, J. P. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. **Autonomous Robots**, Springer, v. 35, n. 1, p. 51–76, 2013. Citado na página 28.

ARANTES, J.; ARANTES, M.; TOLEDO, C.; WILLIAM, B. A multi-population genetic algorithm for uav path re-planning under critical situation. In: **ICTAI**. [S.l.]: IEEE International Conference on Tools with Artificial Intelligence, 2015. p. 1–8. Citado nas páginas 106 e 153.

ARANTES, J. d. S.; ARANTES, M. d. S.; TOLEDO, C. F. M.; JÚNIOR, O. T.; WILLIAMS, B. C. Heuristic and genetic algorithm approaches for uav path planning under critical situation. **International Journal on Artificial Intelligence Tools**, v. 26, n. 01, p. 1760008, 2017. Disponível em: <<http://www.worldscientific.com/doi/abs/10.1142/S0218213017600089>>. Citado nas páginas 106 e 155.

ARANTES, J. S.; ARANTES, M. S.; TOLEDO, C. F. M.; JÚNIOR, O. T.; WILLIAMS, B. C. An embedded system architecture based on genetic algorithms for mission and safety planning with uav. In: **Genetic and Evolutionary Computation Conference**. [S.l.: s.n.], 2017. Citado nas páginas 105 e 157.

ARANTES, M. d. S.; ARIU, K.; TOLEDO, C. F. M.; WILLIAMS, B. C.; ONO, M. Uncertainty improvement for continuous dynamic systems in uav. (**para aparecer**), 2017. (to appear). Citado na página 106.

ARANTES, M. d. S.; TOLEDO, C. F. M.; ARANTES, J.; ARANTES; ONO, M.; WILLIAMS, B. C. Improvements in the probabilistic planning to control continuous stochastic systems. (**para aparecer**), 2017. (to appear). Citado na página 106.

ARANTES, M. d. S.; TOLEDO, C. F. M.; WILLIAMS, B. C.; ONO, M. Collision-free discrete-time encoding for the chance-constrained non convex path planning problem. **IEEE Transactions on Robotics**, 2016. Citado nas páginas 105, 125, 128 e 143.

ARANTES, M. S.; ARANTES, J. S.; TOLEDO, C. F. M.; WILLIAMS, B. C. A hybrid multi-population genetic algorithm for uav path planning. In: **Genetic and Evolutionary Computation Conference**. [S.l.: s.n.], 2016. Citado nas páginas 105 e 147.

ARDUPILOT. **Mission Planner Home**. 2016. <<http://ardupilot.org/planner/index.html>>. Acessado em 30/06/2016. Citado na página 24.

- ARIU, K.; FANG, C.; ARANTES, M.; TOLEDO, C.; WILLIAMS, B. **Chance-Constrained Path Planning with Continuous Time Safety Guarantees**. 2017. Disponível em: <<https://www.aaai.org/ocs/index.php/WS/AAAIW17/paper/view/15149>>. Citado nas páginas 106 e 151.
- BALL, M. O. Heuristics based on mathematical programming. **Surveys in Operations Research and Management Science**, v. 16, n. 1, p. 21 – 38, 2011. ISSN 1876-7354. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1876735410000036>>. Citado na página 31.
- BAYAT, B.; CRASTA, N.; CRESPI, A.; PASCOAL, A. M.; IJSPEERT, A. Environmental monitoring using autonomous vehicles: a survey of recent searching techniques. **Current Opinion in Biotechnology**, v. 45, p. 76 – 84, 2017. ISSN 0958-1669. Energy biotechnology - Environmental biotechnology. Citado na página 25.
- BERG, J. V. D.; ABBEEL, P.; GOLDBERG, K. Lqg-mp: Optimized path planning for robots with motion uncertainty and imperfect state information. **The International Journal of Robotics Research**, p. 895–913, 2011. Citado na página 29.
- BERG, J. V. D.; PATIL, S.; ALTEROVITZ, R. Motion planning under uncertainty using iterative local optimization in belief space. **The International Journal of Robotics Research**, SAGE Publications, v. 31, n. 11, p. 1263–1278, 2012. Citado na página 29.
- BERG, J. van den; PATIL, S.; ALTEROVITZ, R. Motion planning under uncertainty using differential dynamic programming in belief space. In: SPRINGER. **Robotics Research: The 15th International Symposium ISRR**. [S.l.], 2016. v. 100, p. 473. Citado na página 29.
- BESADA-PORTAS, E.; TORRE, L.; MORENO, A.; RISCO-MARTÍN, J. L. On the performance comparison of multi-objective evolutionary {UAV} path planners. **Information Sciences**, v. 238, n. 0, p. 111 – 125, 2013. ISSN 0020-0255. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0020025513001394>>. Citado nas páginas 31 e 32.
- BLACKMORE, L.; ONO, M. Convex chance constrained predictive control without sampling. **Proceedings of the AIAA Guidance, Navigation, and Control Conference**, 2009. Citado nas páginas 25 e 26.
- BLACKMORE, L.; ONO, M.; WILLIAMS, B. C. Chance-constrained optimal path planning with obstacles. **IEEE Transactions on Robotics**, v. 27, n. 6, p. 1080–1094, 2011. Citado nas páginas 19, 21, 25, 26, 27, 28, 29, 30, 43, 44, 46, 52, 53, 93, 106, 112, 115, 117 e 135.
- BLUM, A.; FURST, M. Fast planning through planning graph analysis. **Artificial Intelligence**, 1997. Citado na página 26.
- BLUM, C.; PUCHINGER, J.; RAIDL, G. R.; ROLI, A. Hybrid metaheuristics in combinatorial optimization: A survey. **Applied Soft Computing**, v. 11, n. 6, p. 4135 – 4151, 2011. ISSN 1568-4946. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1568494611000962>>. Citado na página 31.



- BOUSSAÏD, I.; LEPAGNOT, J.; SIARRY, P. A survey on optimization metaheuristics. **Information Sciences**, v. 237, n. 0, p. 82 – 117, 2013. ISSN 0020-0255. Prediction, Control and Diagnosis using Advanced Neural Computations. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0020025513001588>>. Citado na página 30.
- BRY, A.; ROY, N. Rapidly-exploring random belief trees for motion planning under uncertainty. In: IEEE. **Robotics and Automation (ICRA), 2011 IEEE International Conference on**. [S.l.], 2011. p. 723–730. Citado na página 29.
- CULLIGAN, K. F. **Online trajectory planning for uavs using mixed integer linear programming**. Tese (Doutorado) — Massachusetts Institute of Technology, 2006. Citado na página 25.
- DEITS, R.; TEDRAKE, R. Efficient mixed-integer planning for uavs in cluttered environments. **IEEE International Conference on Robotics and Automation**, 2015. Citado nas páginas 28, 29 e 30.
- DEMSAR, J. Statistical comparisons of classifiers over multiple data sets. **J. Mach. Learn. Res.**, JMLR.org, v. 7, p. 1–30, dez. 2006. ISSN 1532-4435. Disponível em: <<http://dl.acm.org/citation.cfm?id=1248547.1248548>>. Citado na página 127.
- EDISON, E.; SHIMA, T. Integrated task assignment and path optimization for cooperating uninhabited aerial vehicles using genetic algorithms. **Computers & Operations Research**, v. 38, n. 1, p. 340 – 356, 2011. ISSN 0305-0548. Project Management and Scheduling. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S030505481000122X>>. Citado nas páginas 31 e 32.
- FREEMAN, P. K.; FREELAND, R. S. Agricultural {UAVs} in the u.s.: potential, policy, and hype. **Remote Sensing Applications: Society and Environment**, v. 2, p. 35 – 43, 2015. ISSN 2352-9385. Citado na página 25.
- GOLDBERG, D. E. **Genetic Algorithms in Search, Optimization and Machine Learning**. 1st. ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989. ISBN 0201157675. Citado na página 31.
- GONDZIO, J. Interior point methods 25 years later. **European Journal of Operational Research**, v. 218, n. 3, p. 587–601, 2012. Disponível em: <<http://EconPapers.repec.org/RePEc:eee:ejores:v:218:y:2012:i:3:p:587-601>>. Citado na página 89.
- GONZÁLEZ, D.; PÉREZ, J.; MILANÉS, V.; NASHASHIBI, F. A review of motion planning techniques for automated vehicles. **IEEE Transactions on Intelligent Transportation Systems**, v. 17, n. 4, p. 1135–1145, April 2016. ISSN 1524-9050. Citado na página 28.
- HASIRCIOGLU, I.; TOPCUOGLU, H. R.; ERMIS, M. 3d path planning for the navigation of unmanned aerial vehicles by using evolutionary algorithms. In: **Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation**. New York, NY, USA: ACM, 2008. (GECCO '08), p. 1499–1506. ISBN 978-1-60558-130-9. Disponível em: <<http://doi.acm.org/10.1145/1389095.1389386>>. Citado na página 31.
- HOFMANN, A. **Robust execution of bipedal walking tasks from biomechanical principles**. Tese (Doutorado) — Massachusetts Institute of Technology, 2006. Citado na página 26.

HOFMANN, A.; WILLIAMS, B. Robust execution of temporally flexible plans for bipedal walking devices. **International Conference on Automated Planning and Scheduling**, 2006. Citado na página 26.

JIN, Y. Surrogate-assisted evolutionary computation: Recent advances and future challenges. **Swarm and Evolutionary Computation**, v. 1, n. 2, p. 61 – 70, 2011. ISSN 2210-6502. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2210650211000198>>. Citado na página 30.

KIRKPATRICK, S.; GELATT, C. D.; VECCHI, M. P. Optimization by simulated annealing. **SCIENCE**, v. 220, n. 4598, p. 671–680, 1983. Citado na página 31.

KUWATA, Y. **Real-time Trajectory Design for Unmanned Aerial Vehicles using Receding Horizon Control**. Dissertação (Mestrado) — Massachusetts Institute of Technology, 2003. Citado na página 25.

LI, H. X. **Kongming: A Generative Planner for Hybrid Systems with Temporally Extended Goals**. Tese (Doutorado) — Massachusetts Institute of Technology, 2010. Citado nas páginas 19, 21, 25, 26, 27, 28, 29, 30, 71, 93, 94, 100 e 131.

LI, H. X.; WILLIAMS, B. C. Hybrid planning with temporally extended goals for sustainable ocean observing. **AAAI Conference on Artificial Intelligence**, 2011. Citado nas páginas 25 e 27.

LUDERS, B. D.; KOTHARI, M.; HOW, J. P. Chance constrained rrt for probabilistic robustness to environmental uncertainty. **AIAA guidance, navigation, and control conference**, 2010. Citado na página 28.

MAC, T. T.; COPOT, C.; TRAN, D. T.; KEYSER, R. D. Heuristic approaches in robot path planning: A survey. **Robotics and Autonomous Systems**, v. 86, p. 13 – 28, 2016. ISSN 0921-8890. Citado na página 28.

MATTMULLER, R.; RINTANEN, J. Planning for temporally extended goals as propositional satisfiability ability. **International Joint Conference on Artificial Intelligence**, 2007. Citado na página 26.

MICHALEWICZ, Z. **Genetic Algorithms + Data Structures = Evolution Programs (3rd Ed.)**. London, UK, UK: Springer-Verlag, 1996. ISBN 3-540-60676-9. Citado na página 31.

NGUYEN, T. T.; YANG, S.; BRANKE, J. Evolutionary dynamic optimization: A survey of the state of the art. **Swarm and Evolutionary Computation**, v. 6, n. 0, p. 1 – 24, 2012. ISSN 2210-6502. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2210650212000363>>. Citado na página 30.

ONO, M. **Robust, Goal-directed Plan Execution with Bounded Risk**. Tese (Doutorado) — Massachusetts Institute of Technology, 2012. Citado nas páginas 28 e 29.

ONO, M.; WILLIAMS, B. C.; BLACKMORE, L. Probabilistic planning for continuous dynamic systems under bounded risk. **Journal of Artificial Intelligence Research**, AI Access Foundation, USA, v. 46, n. 1, p. 511–577, jan. 2013. ISSN 1076-9757. Disponível em: <<http://dl.acm.org/citation.cfm?id=2512538.2512551>>. Citado nas páginas 19, 21, 25, 26, 27, 29, 30, 31, 55, 56, 57, 59, 60, 61, 65, 71, 72, 75, 93, 123 e 125.

- PADEN, B.; ČÁP, M.; YONG, S. Z.; YERSHOV, D.; FRAZZOLI, E. A survey of motion planning and control techniques for self-driving urban vehicles. **IEEE Transactions on Intelligent Vehicles**, IEEE, v. 1, n. 1, p. 33–55, 2016. Citado na página 28.
- RICHARDS, A.; HOW, J. Aircraft trajectory planning with collision avoidance using mixed integer linear programming. **Proceedings of American Control Conference**, 2002. Citado na página 25.
- SALEEM, Y.; REHMANI, M. H.; ZEDADALLY, S. Integration of cognitive radio technology with unmanned aerial vehicles: Issues, opportunities, and future research challenges. **Journal of Network and Computer Applications**, v. 50, p. 15 – 31, 2015. ISSN 1084-8045. Citado na página 25.
- SCHOUWENAARS, T. **Safe trajectory planning of autonomous vehicles**. Tese (Doutorado) — Massachusetts Institute of Technology, 2005. Citado nas páginas 25 e 28.
- SCHOUWENAARS, T.; MOOR, B. D.; FERON, E.; J., H. Mixed integer programming for multi-vehicle path planning. **Proceedings of the European Control Conference**, 2001. Citado na página 25.
- SCUSSEL, A. **Programa aberto de VANTs se torna realidade no Brasil**. 2016. <<http://mundogeo.com/blog/2013/03/04/programa-aberto-de-vants-se-torna-realidade-no-brasil/>>. Acessado em 30/06/2016. Citado na página 22.
- SIERKSMA, G. **Linear and Integer Programming**. 1. ed. New York: Marcel Dekker, 1996. ISBN 0-8247-9695-0. Citado na página 38.
- SUN, W.; BERG, J. van den; ALTEROVITZ, R. Stochastic extended lqr for optimization-based motion planning under uncertainty. **IEEE Transactions on Automation Science and Engineering**, IEEE, v. 13, n. 2, p. 437–447, 2016. Citado na página 29.
- TANG, S.; KUMAR, V. Mixed integer quadratic program trajectory generation for a quadrotor with a cable-suspended payload. **IEEE International Conference on Robotics and Automation**, 2015. Citado nas páginas 29 e 30.
- TOLEDO, C. F. M.; ARANTES, M. S.; OLIVEIRA, R. R. R.; ALMADA-LOBO, B. Glass container production scheduling through hybrid multi-population based evolutionary algorithm. **Applied Soft Computing**, v. 13, n. 3, p. 1352 – 1364, 2013. ISSN 1568-4946. Hybrid evolutionary systems for manufacturing processes. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1568494612001913>>. Citado na página 31.
- TOLEDO, C. F. M.; HOSSOMI, M. Y. B.; ARANTES, M. S.; FRANCA, P. M. Genetic algorithm, mip and improvement heuristic applied to the mlclp with backlogging. In: **Evolutionary Computation (CEC), 2013 IEEE Congress on**. [S.l.: s.n.], 2013. p. 1483–1490. Citado na página 31.
- YANG, X.; WANG, T.; LIANG, J.; YAO, G.; LIU, M. Survey on the novel hybrid aquatic–aerial amphibious aircraft: Aquatic unmanned aerial vehicle (aquauav). **Progress in Aerospace Sciences**, v. 74, p. 131 – 151, 2015. ISSN 0376-0421. Citado na página 25.



---

## MELHORIAS PARA O CCPP

---

O artigo trata do estudo a respeito do CCPP descrito no capítulo 4. O artigo foi submetido em revista internacional com Qualis A1 na ciência da computação e que se encontra na etapa final de revisão na revista. O trabalho apresenta contribuições relevantes ao reformular o CCPP descrito na literatura como um Problema Inteiro Misto (PIM) como um Problema Linear Inteiro Misto (PLIM). Além disso foi possível avançar nos aspectos de escalabilidade e representação. Uma melhor escalabilidade foi alcançada com a proposição de um novo método *Hybrid Incremental Solution Approach* (HISA), que é capaz de resolver cenários com maior número de obstáculos e maior quantidade de waypoints. Avanços na representação foram alcançados ao computar as incertezas que existiam entre os pontos da rota, aspecto não tratado em trabalhos anteriores. Os principais resultados alcançados neste trabalho são sumarizados nas seções 7.1.1 até 7.1.4.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em (ARANTES *et al.*, 2016a)



---

## MELHORIAS PARA O CCQSP

---

O artigo trata do estudo a respeito do CCQSP descrito no capítulo 5 e deverá ser submetido em breve em revista internacional.

Esse trabalho ainda não foi publicado e até a publicação pode ser visto na sua versão atual em <<https://github.com/marcio-da-silva-arantes/PhDThesis/wiki>>





---

## ALGORITMOS EVOLUTIVOS PARA O CCPP

---

---

O trabalho foi publicado em congresso internacional com Qualis A1 na ciência da computação e trata do estudo a respeito do uso de computação evolutiva aplicada ao problema CCPP descrito nessa tese. Os resultados se mostraram promissores ao retornar soluções de boa qualidade em um reduzido tempo computacional. Trata-se de um aspecto relevante, caso o planejador de rota venha a fazer parte do sistema embarcado na aeronave. A seção [7.1.6](#) sumariza este resultado.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em ([ARANTES \*et al.\*, 2016b](#))



---

## ABORDAGENS PARA O DESVIO DE OBSTÁCULOS

---

---

Detalha um artigo que está em fase de elaboração, sendo que foi iniciado durante a visita esse ano ao CSAIL, no MIT, onde a proposta era resolver o problema da alocação do risco, considerando o erro que ocorre quando se discretiza a equação de dinâmica sem levar em consideração o fato de ser uma equação contínua.

Esse trabalho ainda não foi publicado e até a publicação pode ser visto na sua versão atual em <https://github.com/marcio-da-silva-arantes/PhDThesis/wiki>



---

## GARANTIAS NO RISCO DE COLISÃO

---

---

Trata-se de um trabalho em colaboração e publicado em congresso internacional.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em ([ARIU \*et al.\*, 2017](#))



---

## POUSO EM CASO DE FALHAS - CONGRESSO

---

---

Trata-se de um trabalho em colaboração que foi publicado em congresso internacional com Qualis A2 na ciência da computação, discursiva sobre um problema de pouso emergencial no caso do VANT estar impossibilitado de continuar o voo devido a alguma falha.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em ([ARANTES \*et al.\*, 2015](#))





---

## POUSO EM CASO DE FALHAS - REVISTA

---

---

Refere-se a um trabalho em colaboração publicado em revista internacional com Qualis B3 na ciência da computação, discursa sobre um problema de pouso emergencial no caso do VANT estar impossibilitado de continuar o voo devido a alguma falha, trata-se de uma extensão para revista do trabalho apresentado no anexo anterior.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em ([ARANTES \*et al.\*, 2017a](#))



---

## ARQUITETURA EMBARCADA

---

É um trabalho que foi publicado em congresso internacional com Qualis A1 na ciência da computação, expõe estudos no desempenho dos algoritmos para o CCPP quando embarcados numa placa Intel Edson, que possui características passíveis de serem incorporadas em UAVs, pois possuem baixo peso, consumo de energia e processamento comparados a arquiteturas PCs.

Para preservar direitos autorais, esse trabalho não foi anexado na versão revisada da tese, mas pode ser encontrado em ([ARANTES \*et al.\*, 2017b](#))