

**UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA**

FELIPE DESIGLO FERRARE

**OTIMIZAÇÃO DO UAM USANDO MODELAGEM DE VTOL BASEADA
EM SISTEMAS MULTIAGENTES**

São Paulo
2023

FELIPE DESIGLO FERRARE

**OTIMIZAÇÃO DO UAM USANDO MODELAGEM DE VTOL BASEADA
EM SISTEMAS MULTIAGENTES**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do título de Mestre em
Ciências

São Paulo
2023

Nome: FERRARE, Felipe Desiglo

Título: Otimização do UAM Usando Modelagem de VTOL Baseada em Sistemas Multiagentes.

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Ciências.

Aprovado em:

Banca Examinadora

Prof. Dr.

Instituição:

Julgamento:

Prof. Dr.

Instituição:

Julgamento:

Prof. Dr.

Instituição:

Julgamento:

FELIPE DESIGLO FERRARE

**OTIMIZAÇÃO DO UAM USANDO MODELAGEM DE VTOL BASEADA
EM SISTEMAS MULTIAGENTES**

Versão Corrigida

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do título de Mestre em
Ciências

Área de Concentração:

Engenharia de Computação

Orientador:

Prof. Dr. Paulo Sérgio Cugnasca

São Paulo
2023

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Este exemplar foi revisado e corrigido em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, _____ de _____ de _____

Assinatura do autor: _____

Assinatura do orientador: _____

Catálogo-na-publicação

Desiglo Ferrare, Felipe
Otimização do UAM Usando Modelagem de VTOL Baseada em Sistemas Multiagentes / F. Desiglo Ferrare -- versão corr. -- São Paulo, 2023.
184 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo.
Departamento de Engenharia de Computação e Sistemas Digitais.

1.Aeronaves Não Tripuladas 2. Simulação 3. Sistemas Multiagentes
4.Tráfego Aéreo 5.Veículos Autônomos I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II.t.

AGRADECIMENTOS

Agradeço a todos meus amigos e colegas que me apoiaram na minha jornada. Principalmente as várias pessoas com quem tive o prazer de trabalhar junto durante essa jornada, como Derick, Euclides (Neto), Antonio e todos os membros do laboratório do GAS (Grupo de Análise de Segurança) na Poli-USP.

Também a todos os professores da Universidade de São Paulo pela ajuda e colaboração – principalmente ao meu orientador, o Professor Paulo Cugnasca.

“Portanto, também nós temos de nadar e de tentar salvar-nos nessa discussão, ou na esperança que um golfinho nos leve, ou de qualquer outra salvação difícil de conseguir!”

(Platão. República, L. V § 453 d)

RESUMO

Com o uso de UAVs (*Unmanned Aerial Vehicle*), principalmente os veículos aéreos elétricos autônomos pequenos, torna-se cada vez mais importante o gerenciamento adequado e seguro do espaço aéreo. A existência de múltiplos veículos voando em áreas urbanas, permeados por múltiplos obstáculos e outros tipos de tráfego aéreo, faz com que, rapidamente, perceba-se o potencial de problemas e riscos de acidentes e incidentes derivados da implantação dessas novas tecnologias, já que o mecanismo de controle do espaço aéreo, centralizado no ATM (*Air Traffic Management*), e com as regras tradicionais hoje existentes, apresentariam problemas e dificuldades. No contexto dessa mudança de paradigma, novas abordagens sobre o uso do ATM, como o UAM (*Urban Air Mobility*), ganham destaque com a integração do centro urbano com os UAVs pelo uso de meios inteligentes para a definição de rotas. Apesar disso, uma validação em larga escala desse novo e complexo sistema ainda seria difícil e teria seus riscos, o que leva à pertinência do uso de simulação computacional, que é muito mais segura de se realizar do que testes reais, além de apresentar um menor custo. Este trabalho foi desenvolvido tendo como base o que existe atualmente na literatura para o modelamento do funcionamento do UAM, considerando algumas características desse tipo de sistema. Com isso, foi construída uma simulação de alto-nível, utilizando sistemas multiagentes, modelando-se os UAVs como agentes e executando-se diversas simulações em diferentes cenários, buscando-se otimizar as rotas desse novo tipo de tráfego aéreo. Nesse modelo, considerou-se que as aeronaves irão concluir uma missão, que seria o transporte de um objeto ou de uma pessoa, de um ponto a outro dentro do ambiente. Ao final deste trabalho apresenta-se uma análise dos dados obtidos, a utilidade dessas simulações e as conclusões que podem ser tiradas, direcionando para possíveis trabalhos futuros relacionados. Como resultados relevantes pode-se citar que o modelo foi desenvolvido e calibrado com bases nas execuções de simulações, permitindo análise de diversos cenários.

Palavras-chave: Aeronaves Não Tripuladas, Mobilidade Urbana, Simulação, Sistemas Multiagentes, Tráfego Aéreo, Veículos Autônomos.

ABSTRACT

With the use of UAVs (Unmanned Aerial Vehicle), especially small autonomous electric air vehicles, the proper and safe management of airspace becomes increasingly important. The existence of multiple vehicles flying in urban areas, permeated by multiple obstacles and other types of air traffic, makes it possible to quickly perceive the potential for problems and risks of accidents and incidents arising from the implementation of these new technologies, since the airspace control mechanism, focused in the ATM (Air Traffic Management), along with the traditional rules of nowadays, would present problems and difficulties. In the context of this paradigm shift, new approaches to the use of ATM, such as UAM (Urban Air Mobility), become more prominent with the integration of the urban center with the UAVs using intelligent means to define routes. Despite this, a large-scale validation of this new and complex system would still be difficult and would have its risks, which leads to the pertinence of the use of computer simulation, which is much safer to perform than real tests, in addition to presenting a lower cost. This work was developed based on what currently exists in the literature for modeling the operation of the UAM, considering some characteristics of this type of system. With this, a high-level simulation was built, using multi-agent systems, modeling the UAVs as agents and running several simulations in different scenarios, seeking to optimize the routes of this new type of air traffic. In this model, we consider that the aircraft will continue a mission that would be the transport of an object or person from one point to another within the environment. At the end of this work, an analysis of the data obtained is presented along with the usefulness of these simulations and the conclusions that can be drawn towards possible future related works. As relevant results, it can be mentioned that the model was developed and calibrated based on the execution of simulations, allowing the analysis of different scenarios.

Keywords: Air Traffic, Autonomous Vehicles, Multi-Agent Systems, Simulation, Unmanned Aircraft, Urban Mobility.

LISTA DE ILUSTRAÇÕES

Figura 1 – Resumo estado da arte da pesquisa em simulação do UAM com MAS ..	31
Figura 2 – Arquitetura de um Agente	37
Figura 3 – Arquitetura de um MAS	38
Figura 4 – Arquitetura do UAS: entrada de dados externos	42
Figura 5 – Arquitetura do UAS: controle de caminho	43
Figura 6 – Divisão do espaço aéreo tradicional em setores	48
Figura 7 – Aeronave navegando usando uma torre como guia e a sua bússola para seguir o caminho determinado	50
Figura 8 – Carta aeronáutica Rio-São Paulo.....	51
Figura 9 – Detalhe da carta aeronáutica Rio-São Paulo	51
Figura 10 – Detalhe da carta aeronáutica Rio-São Paulo: destaque para o Aeroporto de Congonhas	52
Figura 11 – Divisão de níveis de voo ICAO para voos IFR e VFR	53
Figura 12 – Exemplo de vórtice formado na asa da aeronave	53
Figura 13 – Exemplo de um plano de voo desenhado sobre uma carta aérea	54
Figura 14 – NASA <i>Unmanned Aircraft System (UAS) Traffic Management (UTM)</i> ...	57
Figura 15 – Divisão do espaço aéreo considerada neste trabalho, com destaque à região do UAM	62
Figura 16 – Subdivisão de altitude do UAM considerada.....	63
Figura 17 – Faixas de diferentes altitudes das aeronaves em função de sua altitude	64
Figura 18 – Faixas de diferentes altitudes das aeronaves em função de sua direção	65
Figura 19 – Linha de 1000 ft acima do solo	66
Figura 20 – Mapa da cidade de São Paulo e macrorregiões correlatas, em escala..	67
Figura 21 – Áreas quadradas de simulação sobre o mapa da cidade de São Paulo e sobre a macrorregião da cidade São Paulo	68
Figura 22 – Representação esquemática de uma quadrícula	69
Figura 23 – Etapa 1: Aeronave antes da partida	70
Figura 24 – Etapa 2: Bloqueio de quadrículas antes da subida	71
Figura 25 – Etapa 3: Desbloqueio de quadrículas durante a subida	72
Figura 26 – Etapa 4: Aeronave atinge altitude alvo.....	72
Figura 27 – Etapa 5: Navegação horizontal normal da aeronave.....	73
Figura 28 – Etapa 6: Bloqueio de quadrículas para a descida	73
Figura 29 – Etapa 7: Desbloqueio de quadrículas durante a descida	74

Figura 30 – Etapa 8: Aeronave pousa no seu destino.....	74
Figura 31 – Desvio de obstáculo pelo lado direito.....	76
Figura 32 – Desvio de obstáculo por baixo	77
Figura 33 – Janela do Netlogo e Interface do Modelo.....	78
Figura 34 – Detalhes Gráficos da Interface do Modelo	79
Figura 35 – Detalhe dos resultados numéricos da interface do modelo.....	79
Figura 36 – Janela de simulação do Netlogo	80
Figura 37 – Ambiente de simulação vazio (sem aeronaves).....	81
Figura 38 – Ambiente de simulação com aeronaves (agentes): em vermelho VTOLs do tipo 1 e, em amarelo, VTOLs do tipo 2.....	82
Figura 39 – Vista superior do ambiente de simulação com múltiplas aeronaves (agentes).....	82
Figura 40 – Vista lateral do ambiente com agentes	83
Figura 41 – Estrutura de Dados do Agente “Aeronave”.....	86
Figura 42 – Estrutura de Dados da “Quadrícula”.....	87
Figura 43 – Exemplo de código de geração de agentes	87
Figura 44 – Diagrama dos diferentes tipos de CBs (CB-1 e CB-2)	89
Figura 45 – CB Comprida ao lado de uma aeronave	89

LISTA DE TABELAS

Tabela 1 – Resumo: Trabalhos Relacionados.....	29
Tabela 2 – Diagrama simplificado do modelo, utilizando o protocolo ODD.....	103
Tabela 3 – Modelo ODD.....	104
Tabela 4 – Diagrama completo do modelo, utilizando o protocolo ODD.....	104
Tabela 5 – Configuração de parâmetros do primeiro exemplo: Número de agentes e ciclo.....	108
Tabela 6 – Configuração de parâmetros do primeiro exemplo: Separação.....	108
Tabela 7 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 1.....	108
Tabela 8 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 2.....	109
Tabela 9 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 3.....	109
Tabela 10 – Execuções da simulação: Cenário 1 (parte 1).....	109
Tabela 11 – Execuções da simulação: Cenário 1 (parte 2).....	110
Tabela 12 – Resumo dos resultados do Cenário 1 (parte 1).....	111
Tabela 13 – Resumo dos resultados do Cenário 1 (parte 2).....	111
Tabela 14 – Resumo dos resultados do Cenário 2 (parte 1).....	112
Tabela 15 – Resumo dos resultados do Cenário 2 (parte 2).....	112
Tabela 16 – Resumo dos resultados do Cenário 3 (parte 1).....	113
Tabela 17 – Resumo dos resultados do Cenário 3 (parte 2).....	113
Tabela 18 – Velocidades dos VTOLs – Caso 1.....	117
Tabela 19 – Velocidades dos VTOLs – Caso 2.....	117
Tabela 20 – Velocidades dos VTOLs – Caso 3.....	117
Tabela 21 – Trajetória Paralela: Resultados do Cenário 4 – Caso 1 (parte 1).....	118
Tabela 22 – Trajetória Paralela: Resultados do Cenário 4 – Caso 1 (parte 2).....	119
Tabela 23 – Trajetória Paralela: Resultados do Cenário 5 – Caso 2 (parte 1).....	119
Tabela 24 – Trajetória Paralela: Resultados do Cenário 5 – Caso 2 (parte 2).....	120
Tabela 25 – Trajetória Paralela: Resultados do Cenário 6 – Caso 3 (parte 1).....	120
Tabela 26 – Trajetória Paralela: Resultados do Cenário 6 – Caso 3 (parte 2).....	121
Tabela 27 – Trajetória Paralela: Resultados do Cenário 7 – Caso 2 com CB-1 (parte 1).....	121
Tabela 28 – Trajetória Paralela: Resultados do Cenário 7 – Caso 2 com CB-1 (parte 2).....	122

Tabela 29 – Trajetória Paralela: Resultados do Cenário 8 – Caso 2 com CB-2 (parte 1).....	122
Tabela 30 – Trajetória Paralela: Resultados do Cenário 8 – Caso 2 com CB-2 (parte 2).....	123
Tabela 31 – Trajetória Perpendicular: Resultados do Cenário 9 – Caso 1 (parte 1).....	124
Tabela 32 – Trajetória Perpendicular: Resultados do Cenário 9 – Caso 1 (parte 2).....	125
Tabela 33 – Trajetória Perpendicular: Resultados do Cenário 10 – Caso 2 (parte 1).....	125
Tabela 34 – Trajetória Perpendicular: Resultados do Cenário 10 – Caso 2 (parte 2).....	126
Tabela 35 – Trajetória Perpendicular: Resultados do Cenário 11 – Caso 3 (parte 1).....	126
Tabela 36 – Trajetória Perpendicular: Resultados do Cenário 11 – Caso 3 (parte 2).....	127
Tabela 37 – Trajetória Perpendicular: Resultados do Cenário 12 – Caso 2 com CB-1 (parte 1).....	127
Tabela 38 – Trajetória Perpendicular: Resultados do Cenário 12 – Caso 2 com CB-1 (parte 2).....	128
Tabela 39 – Trajetória Perpendicular: Resultados do Cenário 13 – Caso 2 com CB-2 (parte 1).....	128
Tabela 40 – Trajetória Perpendicular: Resultados do Cenário 13 – Caso 2 com CB-2 (parte 2).....	129
Tabela 41 – Trajetória Geral: Resultados do Cenário 14 – Caso 1 (parte 1).....	130
Tabela 42 – Trajetória Geral: Resultados do Cenário 14 – Caso 1 (parte 2).....	131
Tabela 43 – Trajetória Geral: Resultados do Cenário 15 – Caso 2 (parte 1).....	131
Tabela 44 – Trajetória Geral: Resultados do Cenário 15 – Caso 2 (parte 2).....	132
Tabela 45 – Trajetória Geral: Resultados do Cenário 16 – Caso 3 (parte 1).....	132
Tabela 46 – Trajetória Geral: Resultados do Cenário 16 – Caso 3 (parte 2).....	133
Tabela 47 – Trajetória Geral: Resultados do Cenário 17 – Caso 2 com CB-1 (parte 1).....	133
Tabela 48 – Trajetória Geral: Resultados do Cenário 17 – Caso 2 com CB-1 (parte 2).....	134
Tabela 49 – Trajetória Geral: Resultados do Cenário 18 – Caso 2 com CB-2 (parte 1).....	134
Tabela 50 – Trajetória Geral: Resultados do Cenário 18 – Caso 2 com CB-2 (parte 2).....	135
Tabela 51 – Lista de Cenários de Separação.....	136
Tabela 52 – Separação: Resultados do Cenário 19 (parte 1).....	136
Tabela 53 – Separação: Resultados do Cenário 19 (parte 2).....	137

Tabela 54 – Separação: Resultados do Cenário 20 (parte 1)	137
Tabela 55 – Separação: Resultados do Cenário 20 (parte 2)	138
Tabela 56 – Lista de Cenários de Quantidade de Aeronaves	139
Tabela 57 – Quantidade de VTOL-1: Resultados do Cenário 21 – Caso 1 (parte 1)	139
Tabela 58 – Quantidade de VTOL-1: Resultados do Cenário 21 – Caso 1 (parte 2)	140
Tabela 59 – Quantidade de VTOL-1: Resultados do Cenário 22 – Caso 1 (parte 1)	140
Tabela 60 – Quantidade de VTOL-1: Resultados do Cenário 22 – Caso 1 (parte 2)	141
Tabela 61 – Quantidade de VTOL-1: Resultados do Cenário 23 – Caso 2 (parte 1)	141
Tabela 62 – Quantidade de VTOL-1: Resultados do Cenário 23 – Caso 2 (parte 2)	142
Tabela 63 – Quantidade de VTOL-1: Resultados do Cenário 24 – Caso 2 (parte 1)	142
Tabela 64 – Quantidade de VTOL-1: Resultados do Cenário 24 – Caso 2 (parte 2)	143

LISTA DE SIGLAS E ABREVIações

ANAC	Agência Nacional de Aviação Civil
ANATEL	Agência Nacional de Telecomunicações
ATC	<i>Air Traffic Control</i>
ATCo	<i>Air Traffic Controller</i>
ATM	<i>Air Traffic Management</i>
BVLOS	<i>Beyond Visual Line of Sight</i>
CB	Cumulonimbus
CTOL	<i>Conventional Take-Off and Landing</i>
DAI	<i>Distributed Artificial Intelligence</i>
DECEA	Departamento de Controle do Espaço Aéreo
EASA	<i>European Aviation Safety Agency</i>
EVLOS	<i>Extended Visual Line Of Sight</i>
eVTOL	<i>electric Vertical Take-Off and Landing</i>
FAA	<i>Federal Aviation Administration</i>
IA	Inteligência Artificial
ICAO	<i>International Civil Aviation Organization</i>
IFR	<i>Instrument Flight Rules</i>
MAS	<i>Multiple Agent Systems</i>
mini-UAVs	<i>mini Unmanned Aerial Vehicles</i>
MN	Milha Náutica
nano-UAVs	<i>nano Unmanned Aerial Vehicles</i>
NASA	<i>National Aeronautics and Space Administration</i>
ODD	<i>Overview, Design Concepts, Details</i>
RPAS	<i>Remotely Piloted Aircraft Systems</i>
UAM	<i>Urban Air Mobility</i>
UAS	<i>Unmanned Aircraft System</i>
UAV	<i>Unmanned Aerial Vehicle</i>
UPS	<i>United Parcel Service</i>
UTM	<i>Unmanned Traffic Management</i>
VANT	Veículo Aéreo Não Tripulado
VLOS	<i>Visual Line Of Sight</i>
VRF	<i>Visual Flight Rules</i>
VTOL	<i>Vertical Take-Off and Landing</i>

SUMÁRIO

1. INTRODUÇÃO	17
1.1 MOTIVAÇÃO	21
1.2 OBJETIVOS	23
1.3 JUSTIFICATIVA	23
1.4 CONTRIBUIÇÕES ESPERADAS	24
1.5 METODOLOGIA	25
1.6 ESTRUTURA DO TRABALHO	25
2. TRABALHOS RELACIONADOS	27
3. SISTEMAS MULTIAGENTES	32
3.1 AGENTES	32
3.2 MULTIAGENTES	33
3.3 LINGUAGEM NETLOGO	38
4. UNMANNED AIRCRAFT SYSTEMS (UAS)	40
4.1 UNMANNED AERIAL VEHICLES (UAVs)	40
4.2 AIR TRAFFIC MANAGEMENT (ATM)	45
4.3 UAS TRAFFIC MANAGEMENT (UTM) E URBAN AIR MOBILITY (UAM)	55
5. MODELO MULTIAGENTE PARA SIMULAÇÃO DE UAVS	59
5.1 CONSIDERAÇÕES GERAIS DO MODELO	59
5.2 DESCRIÇÃO DO COMPORTAMENTO DO MODELO	69
5.3 INTERFACE DE CONTROLE DO NETLOGO	77
5.4 INTERFACE DE SIMULAÇÃO DO NETLOGO	80
5.5 AGENTES CRIADOS NO NETLOGO	83
5.6 INTERFACE DOS AGENTES NO NETLOGO	86
5.7 CUMULONIMBUS (CBs)	87
5.8 CENÁRIOS DE TRAJETÓRIAS DE AERONAVES	92
5.9 PARÂMETROS DO MODELO MULTIAGENTE	96
5.10 SEPARAÇÃO MÍNIMA E SEPARAÇÃO ESTIMULADA	99
5.11 ESTADO DE ESPERA DE AERONAVES (ESTADO DE <i>DELAY</i>)	100

5.12	DEFINIÇÃO DO MODELO PELO PROTOCOLO ODD	101
6.	RESULTADOS	107
6.1	PRIMEIRO EXEMPLO.....	107
6.2	DIFERENTES TIPOS DE CENÁRIOS	117
6.3	RESULTADOS GERAL – SEPARAÇÃO.....	135
6.4	RESULTADOS GERAIS – QUANTIDADE DE AERONAVES	139
6.5	ANÁLISES GERAIS.....	143
7.	CONCLUSÃO.....	146
7.1	CONTRIBUIÇÕES.....	147
7.2	TRABALHOS FUTUROS	147
7.3	CONSIDERAÇÕES FINAIS	148
	REFERÊNCIAS.....	150
	APÊNDICE A – DESCRIÇÃO DETALHADA DO MODELO.....	157
	APÊNDICE B – CÓDIGO FONTE DO MODELO NETLOGO DESENVOLVIDO....	162
	APÊNDICE B.1 DEFINIÇÃO DAS ENTIDADES.....	163
	APÊNDICE B.2 VARIÁVEIS GLOBAIS.....	164
	APÊNDICE B.3 <i>SETUP</i> DE VARIÁVEIS.....	166
	APÊNDICE B.4 CONFIGURAÇÕES DE CB	168
	APÊNDICE B.5 BLOQUEIOS, DESVIOS, ETC.....	170
	APÊNDICE B.6 EXECUÇÃO DO MODELO	172
	APÊNDICE B.7 GERAÇÃO E CONFIGURAÇÃO DE AERONAVES	173
	APÊNDICE B.8 MOVIMENTAÇÃO DE AERONAVES	176
	APÊNDICE B.9 EXECUÇÃO DO <i>BATCH</i>	182

1. INTRODUÇÃO

Os UAVs – *Unmanned Aerial Vehicles* (em português, VANTs – Veículos Aéreos Não Tripulados), também popularmente conhecidos como *drones*, representam uma classe genérica de veículos aéreos que inclui desde veículos remotamente pilotados (RPAS – *Remotely Piloted Aircraft Systems*) até veículos totalmente autônomos. Além disso, por meio de uma revisão na literatura sobre tais tipos de veículos, pode-se destacar algumas classes (tipos), como os veículos de decolagem horizontal (CTOL – *Conventional Take-Off and Landing*) e os veículos de decolagem vertical (VTOL – *Vertical Take-Off and Landing*), sendo que, dentre esses últimos, constituem-se entre os mais importantes os assim chamados quadricópteros. Esses tipos de veículos existem em uma grande variedade de tamanhos, desde aeronaves menores que 250g (nano-UAVs), passando por aeronaves menores que 25 kg (*small-UAVs*), indo até aeronaves de tamanho convencional (Shakhathreh et al., 2018).

Apesar dessa diversidade de peso e tamanho, cada vez mais são necessárias soluções e regulamentos capazes de englobar todos esses tipos de aeronaves, incluindo as aeronaves hoje existentes, mantendo-se os níveis de segurança e confiabilidade atuais referente ao espaço aéreo. Não é de hoje que se tem pensado na construção de aeronaves autônomas. Desde o início da aviação existe um esforço da engenharia para a criação de aeronaves cada vez mais autônomas e, nos últimos anos, tem ocorrido uma grande ênfase no desenvolvimento dessa tecnologia.

Atualmente, o desenvolvimento dos UAVs ultrapassa o escopo das aeronaves em si e alcança, também, os diversos sistemas necessários para a sua operação e a comunicação entre aeronaves dessa natureza com outras entidades constituintes do espaço aéreo e do controle do espaço aéreo. Esse conjunto também é conhecido como UASs (*Unmanned Aircraft Systems*). Dessa forma, dá-se um destaque não só para o desenvolvimento de tecnologias voltadas para as aeronaves, mas também de tecnologias direcionadas para os sistemas envolvidos na sua operação, incluindo aqueles localizados em solo e os que envolvem múltiplas aeronaves, como os sistemas de controle e de gerenciamento de tráfego aéreo – ATM (*Air Traffic Management*).

Já se considera que os UAVs terão um grande impacto em múltiplos setores da economia, como no transporte de passageiros, nos serviços de entrega de encomendas, etc., gerando grandes mudanças nessas áreas (BALLVE, 2014). Apesar de haver uma longa história dessa área, datando desde o início da aviação no começo do século XX, nos últimos anos muitas dessas mudanças têm ocorrido com o avanço das tecnologias que têm permitido implementar essas inovações em ambientes reais (Cohen et al., 2021). Além disso, atualmente essas tecnologias mostram-se extremamente úteis em áreas como a agricultura, com um mercado estimado em 32,4 bilhões dólares pela empresa de consultoria/auditoria PwC (PricewaterhouseCoopers), além de serviços de inspeção, manutenção e segurança (JOSHI, 2017) (Fahmani et al., 2020). Assim, tem-se visto cada vez mais aplicações e novos usos envolvendo os UAVs, bem como um crescente desenvolvimento e investimento em tecnologia nessa área.

Grandes empresas da área de logística, como a Amazon e a UPS (*United Parcel Service*) têm investido nos últimos anos para criar maneiras de viabilizar sistemas capazes de fazer entregas em massa, realizando todo o percurso entre o centro de entregas até a residência do consumidor (DESJARDINS, 2018). Já empresas como a Joby aviation (antiga divisão de UAVs da Uber) têm desenvolvido soluções como os carros voadores, voltados para o transporte de passageiros e a serem colocados em prática nos grandes centros urbanos em um futuro próximo (HAWKINS, 2018). A Uber colocou a sua divisão de pesquisa em veículos terrestres (carros) e aéreos (VTOL) autônomos à venda em 2020 devido a dificuldades financeiras surgidas durante a pandemia do COVID-19, mas ela continua mantendo uma participação na empresa resultante e pretende manter os planos de implementação dos assim chamados “carros voadores” (BBC, 2021). Várias regiões do planeta, como a cidade de Dubai, já possuem planos de desenvolvimento de redes de “táxis-voadores” (Wakefield, 2017). No Brasil, a Embraer abriu capital da sua divisão Eve na bolsa de Nova Iorque e pretende colocar esses tipos de veículos em operação entre 2025 e 2026 (IG, 2022).

Em outros lugares, como em Tóquio, existia um projeto de se ter os primeiros testes com “carros voadores” (como tem-se chamado os UAVs voltados para transporte de passageiros) para a Olimpíada de 2020 – posteriormente postergado para 2021 com a chegada da pandemia do COVID-19 –, e que tem como meta, até 2050, a criação de um sistema de transporte totalmente operante na cidade inteira

(Demetriou, 2017). Esse projeto, apesar de atrasado pela pandemia, foi apresentado pela Toyota na forma de um modelo (protótipo) de veículos aéreos de transporte de passageiros para acender a tocha olímpica em 2021, mostrando 4 minutos de voo com um passageiro; apesar de tudo, esse protótipo ainda necessita utilizar um piloto humano na aeronave (G1, 2020). A Olimpíada ocorreu em 2021 sem público por causa da pandemia do COVID-19, e a tocha foi acesa por uma pessoa (Metrópolis, 2021). Adicionalmente, foi realizado um *show* com a utilização de 1.824 *drones* que formavam, dinamicamente, diversas figuras e desenhos no céu (Época Negócios, 2021).

Para viabilizar tudo isso, são necessárias mudanças em diversas regulamentações e leis, além de projetos de educação do público para poder se chegar a regras claras e padronizadas de como deve ser a operação desses veículos, de maneira a garantir a segurança do sistema de tráfego aéreo, das pessoas e propriedades, visando-se evitar acidentes e incidentes. Várias normas têm sido criadas no Brasil pela ANATEL (Agência Nacional de Telecomunicações) para regulamentar a comunicação e o uso do espaço aéreo pelo DECEA (Departamento de Controle do Espaço Aéreo) e pela ANAC (Agência Nacional de Aviação Civil), e isso tem acontecido também em diversos países da Europa e dos Estados Unidos da América, mas as regras atuais ainda são limitadas e há muito o que se desenvolver a respeito (ANAC, 2017).

Contudo, com a diminuição dos custos dessas novas tecnologias e com a sua popularização, tem ocorrido o aumento de problemas, como invasões de *drones* em áreas restritas e prejuízos em resgates em áreas de desastres (G1, 2019). Essas ações podem colocar determinados eventos em risco, forçando organizadores e órgãos do governo a investir em tecnologias de controle e monitoramento (O'KANE, 2019).

Além disso, a introdução de *drones* próximos ao tráfego aéreo hoje existente tem causado riscos de colisões com as aeronaves atuais, e, por diversas vezes, os *drones* chegam a invadir áreas de aeroportos. Isso causa o fechamento destes, gerando inúmeros problemas para o gerenciamento do tráfego aéreo (HOLLISTER, 2019) (STAFF, 2018) (GUARDIAN, 2019).

Cada vez mais, a discussão legal a esse respeito tem se aprofundado sobre quais áreas podem ser acessadas pelos *drones*, incluindo-se aí veículos aéreos de forças policiais, veículos voando muito baixo em áreas particulares, ou veículos

aéreos realizando filmagens de áreas (mesmo sem sobrevoar), pois criam-se situações que podem violar a privacidade das pessoas. Com isso, novas regras e decisões judiciais aumentam a discussão sobre o que pode e o que não pode ser feito nesses casos (Blank et al., 2018).

Como propostas para a evolução do ATM (*Air Traffic Management*), tem-se o advento do UTM (*Unmanned Traffic Management*) da NASA (*National Aeronautics and Space Administration*) para o controle do tráfego aéreo em ambientes segregados (onde só existam UAVs), ou propostas mais complexas de repensar a mobilidade urbana como um todo como o *Urban Air Mobility* (UAM). Nesse último caso, o UAM faria o controle das trajetórias e das áreas de voo restrito e/ou proibido de maneira dinâmica, de forma que todas as aeronaves poderiam acessar dados relacionados a isso e, assim, evitar acidentes e infrações em espaços aéreos restritos (HOMOLA et al., 2016).

Outro aspecto desenvolvido na literatura é a integração de serviços utilizando UAVs com as *smart cities* (cidades inteligentes) (Giyenko e Cho, 2016), acarretando o desenvolvimento do UAM. Esse desenvolvimento consiste na integração da gestão de tráfego de UAVs, em ambientes de alta densidade populacional, com o funcionamento do tráfego aéreo da cidade (outros usuários do espaço aéreo), considerando falhas de sistemas e mudança climáticas, além de restrições e mudanças na dinâmica do controle do espaço aéreo quando ocorrem emergências e necessidades de uso do espaço aéreo por parte do governo (Thipphavong et al., 2019).

Assim, a introdução de UAVs no espaço aéreo atual poderia envolver questões de reserva de áreas autorizadas de voo para esse fim, bem como poderia gerar novas regras de gerenciamento do espaço aéreo, que estaria mudando dinamicamente, em questão de minutos ou horas, em um contexto diferente da aviação tradicional, na qual os voos são planejados com dias e até meses de antecedência. Além disso, a introdução de UAVs nos grandes centros urbanos traria uma quantidade muito maior de aeronaves podendo voar naquele espaço aéreo, de alguns centímetros do solo até alturas compatíveis com a aviação tradicional, o que potencialmente geraria inúmeros desafios e problemas.

Pode-se acrescentar que, com a integração de diferentes agentes, empresas e órgãos públicos no sistema aéreo, operando conjuntamente, trazem à tona várias questões do funcionamento da cidade e integração de diferentes serviços, como

serviços de entregas ou de transporte operados por empresas concorrentes, além de outros indivíduos que queiram operar aeronaves, como aeronaves de serviços públicos operados pelo próprio governo, que podem ser operacionalizados por meio do uso de UAVs. Assim é necessário um sistema para mediar esses interesses conflitantes.

Apesar disso, existem ainda poucas regras definidas, muitos problemas decorrentes e múltiplos desafios. Mesmo com o grande potencial de crescimento da utilização de UAVs no espaço aéreo, tem-se ainda muitos desafios e variáveis a serem aprendidas para se poder otimizar o sistema de controle do espaço aéreo com segurança.

1.1 MOTIVAÇÃO

Considerando os cenários expostos, tem-se algumas situações complexas e difíceis de se ter em um ambiente real, mantendo-se os níveis de segurança, e não colocando pessoas e bens em risco. Além disso, são relatadas na literatura muitas soluções propostas para o contexto do UAM e para o uso de VTOLs em geral, de maneira teórica, sem resultados para validá-las e apenas utilizando comparativos com cenários pequenos ou com a aviação convencional, como em Lei (2011), que gera cenários com apenas uma aeronave, ou em Wei (2015), que concebe um modelo para auxiliar o sistema de controle de tráfego aéreo tradicional.

A operação de eVTOL (*electric VTOL*), um tipo particular de UAVs, apresenta uma série de particularidades, principalmente em ambientes urbanos, que tendem a ser congestionados e possuir muitos gargalos. Além disso, não é possível utilizar-se de dados reais de operação para este tipo de análise, podendo esta abordagem com o uso de simulação ser utilizada não só para o UAS, como também em outras áreas da aviação, como observa-se em Neto (2022). Assim, a utilização de simulações torna-se um ponto viável para validação de múltiplos cenários e teste de novos modelos relacionados com essa problemática. Desta forma, neste trabalho propõe-se um modelo, validado por meio de simulações computacionais, para comprovar tais resultados.

As novas tecnologias têm que abarcar o controle de múltiplos veículos se deslocando e considerando as regras definidas do tráfego aéreo atual em uma dada região, que podem mudar em questão de minutos, devido à ocorrência de eventos

inesperados, como acidentes; a pousos de emergência e a deslocamentos de autoridades, que podem forçar todo um espaço aéreo a ficar com acesso restrito; ou a incêndios (fumaça), chuva e ventos, que podem prejudicar o deslocamento das aeronaves e, assim, forçar desvios de suas rotas ou o aumento dos limites operacionais de segurança. Por isso, é necessário que, além de existir uma troca de informações entre os veículos envolvidos – comunicando-se, assim, suas intenções de voo e manobras –, que as entidades reguladoras de tráfego aéreo também colem essas informações para monitorar o sistema e colaborem com dados sobre clima e disponibilidade de acesso ao espaço aéreo.

Os desenvolvimentos tecnológicos necessários não só se restringem ao setor aéreo, como existe um esforço generalizado nos últimos anos por automação e a criação de veículos autônomos de tipos diversos, desde trens, carros, navios e submarinos, colaborando com um interesse cada vez maior na área de veículos autônomos em geral. Apesar das particularidades técnicas específicas de cada uma dessas iniciativas, verifica-se o grande interesse nessas aplicações visando sempre os mesmos objetivos, como reduzir a mão-de-obra, reduzir gastos de energia, otimizar a execução de tarefas e automatizar tarefas potencialmente perigosas, como em transporte de passageiros (Maget, 2020), serviços de entregas de pacotes (Reiche, 2019) e até em inspeções técnicas (Montambault, 2010) e segurança pública (Ashish, 2020). Pode-se concluir, a partir disso, que os trabalhos nessa área não se mostram só em grande destaque, como também representam uma necessidade social mais ampla com impacto no futuro das pessoas.

Muitos trabalhos focam no uso de aeronaves autônomas, apresentando resultados de pesquisa envolvendo apenas uma aeronave autônoma (ou agente). Esses cenários não se encaixam com os cenários futuros a longo prazo, em que o espaço aéreo vai ser dividido entre múltiplas aeronaves autônomas (ou agentes). Destaca-se, ainda, que apesar do uso de MAS (*Multi-Agent Systems*) não ser a única solução possível para esse problema, ela se mostra uma solução com grande potencial para resolver problemas desta natureza, como se pode observar em Lei (2011) e Wei (2015), que utilizam sistemas multiagentes na área de aviação.

1.2 OBJETIVOS

O objetivo principal deste trabalho de pesquisa é propor um modelo de simulação de Veículos Aéreos Não Tripulados (VANTs) do tipo VTOL (*Vertical Take-Off and Landing*), baseado em sistemas multiagentes, aplicáveis a ambientes de gerenciamento do espaço aéreo denominado UAM (*Urban Air Mobility*), no qual os VTOLs são os agentes e o UAM (e suas regras) correspondem o ambiente onde os agentes estão localizados.

Ainda, como objetivos secundários deste trabalho, pretende-se:

- Propor um modelo multiagentes do UAM;
- Validar as trajetórias dos VTOLs quanto à segurança e à otimização do sistema aéreo;
- Gerar resultados para possibilitar a avaliação de índices de complexidade/capacidade do UAM.

1.3 JUSTIFICATIVA

Esta pesquisa justifica-se porque, devido à alta complexidade dos sistemas de gestão de tráfego aéreo, o desenvolvimento necessário para o aprimoramento do ATM para a acomodação desse novo tipo de tráfego, incluindo VTOLs, faz-se necessário. O sistema atual não possui capacidade nem regras específicas para conseguir lidar com esse novo tráfego de aeronaves. Então, torna-se necessária a migração para um sistema de controle específico para esses veículos como o UAM.

Com as dificuldades de gerenciamento do ATM, já que a capacidade de controle de um operador humano é limitada a poucas aeronaves, ao se colocar mais aeronaves, ainda que autônomas, isso necessariamente implicaria a contratação de mais operadores. No caso da introdução dos UAVs, essa operação, além de não garantir a segurança, não é uma opção economicamente viável. Esse trabalho não visa dar uma alternativa para substituir o ATM, mas sim proporcionar um modelo de extensão do ATM, com a introdução do UAM. Com isso, espera-se que, no futuro, o modelo proposto para o UAM possa ser acrescido para seu uso em harmonia com o ATM, acarretando em um modelo híbrido entre o que existe atualmente (ATM) e o UAM.

Apesar de propostas como o UTM da NASA já existirem, elas não mostram quais regras são utilizadas para o controle dos UAVs e como o sistema deverá ser operado. No atual momento, os UAVs já são usados em grande quantidade, com regras variadas, e sem muito controle, tornando essa prática ineficiente e insegura.

Considerando o problema exposto e que existem para ele poucas soluções, deixar para validar ou analisar os possíveis cenários em uma situação real mostra-se pouco prático, além de perigoso. O uso de simulações computacionais, principalmente para validar cenários mais difíceis de testar, como a ocorrência de mau tempo ou a existência de grande quantidade de veículos ou a execução de testes por horas seguidas, torna-se mais seguro, barato e ágil.

Para manter os níveis de segurança e permitir os novos desenvolvimentos emergentes na área dos UAVs, faz-se necessário o desenvolvimento de novos sistemas de controle mais inteligentes e autônomos. Apesar da importância do tema e da ampla gama de trabalhos surgidos nesta área, existe a possibilidade de um aprofundamento maior, principalmente com foco onde existem múltiplos veículos; ainda, medir a confiabilidade do sistema e definir as regras de interação entre esses veículos são aspectos que merecem atenção no desenvolvimento de tais sistemas.

1.4 CONTRIBUIÇÕES ESPERADAS

Neste trabalho, pretende-se apresentar, como principal contribuição, a proposta de modelo para o UTM utilizando sistemas multiagentes em ambientes segregados (podendo ser estendido, futuramente, para ambientes não segregados), capaz de simular condições de trajetórias, encadeamento de agentes, além de situações de congestionamento e de clima adverso.

Para isso, faz-se necessária a construção de um simulador que pode ser utilizado para múltiplas validações, tanto de regras referentes ao espaço aéreo controlado, quanto de confiabilidade e capacidade do UAM.

Espera-se, dessa forma, contribuir com o avanço do estado da arte, por meio de um modelo e resultados de simulações, que podem ser úteis para um futuro UTM ou UAM, os quais podem ser utilizados para validar regras e parâmetros de capacidade do espaço aéreo controlado, bem como gerar parâmetros gerais de uso de ambientes simulados, usando diferentes parâmetros de segurança e desempenho de agentes.

1.5 METODOLOGIA

Seguindo a metodologia descrita na literatura (SILVEIRA; CÓRDOVA, 2009) (WAZLAWICK, 2009), este trabalho foi desenvolvido na forma de uma pesquisa quantitativa, exploratória e experimental. Quantitativa, pois foi desenvolvida utilizando-se de modelos, com o uso de simulações, para o seu desenvolvimento. E sendo pesquisa exploratória, foi feita uma busca bibliográfica e análise de exemplos para o desenvolvimento do trabalho (SILVEIRA; CÓRDOVA, 2009). E sendo também pesquisa experimental, busca-se embasar os seus objetivos em conclusões fundamentadas por experimentos (ou simulações) elaborados durante a pesquisa (SILVEIRA; CÓRDOVA, 2009):

Nesse modelo são considerados como variáveis o número de agentes (UAVs), a velocidade das aeronaves e o espaço restrito para os voos. São avaliadas as possíveis melhorias de desempenho no cenário utilizando como referência: nível de segurança do sistema, distância entre aeronaves, tempo de voo de cada aeronave e tempo de resposta do sistema. Para a execução dessas atividades, este projeto de pesquisa encontra-se dividido em cinco fases:

1. Fase de análise e revisão bibliográfica;
2. Fase de coleta de dados e de construção do modelo (modelagem);
3. Fase de desenvolvimento de cenários de testes e simulação;
4. Fase de consolidação e análise dos dados (resultados);
5. Fase de síntese e conclusão.

1.6 ESTRUTURA DO TRABALHO

No capítulo 1, foi feita a introdução sobre o trabalho, contextualizando a situação atual e destacando os principais problemas e acontecimentos recentes na área. Também são apresentados os objetivos deste trabalho de pesquisa, a sua motivação e justificativa do tema, com as contribuições geradas para o estado da arte.

No capítulo 2, encontra-se a revisão de literatura, em que são listados os principais trabalhos sendo desenvolvidos na área e suas respectivas contribuições; assim, pode-se ter uma visão geral de onde este trabalho se situa no contexto das pesquisas atuais.

No capítulo 3, é feita uma contextualização sobre sistemas multiagentes, tendo como objetivo a apresentação geral do tema. Também são tratados alguns conceitos básicos, terminologias e suas aplicações para a indústria da aviação e de UAVs.

No capítulo 4, apresentam-se definições sobre o UAS e o UAM. Da mesma forma que no capítulo anterior, são apresentados conceitos básicos, terminologias e seus problemas, bem como questões relativas à aviação tradicional e ao desenvolvimento relacionado com a criação de trajetórias em ambientes com UAS e seus desafios.

No capítulo, 5 é apresentado o modelo desenvolvido neste trabalho de pesquisa de Mestrado. Durante esse capítulo é mostrado, de maneira detalhada, como o modelo foi desenvolvido, suas principais características, como foram escolhidos os parâmetros de simulação e quais foram as decisões tomadas durante o desenvolvimento do modelo.

No capítulo 6, compilam-se os resultados obtidos e realiza-se uma análise desses itens. Foram selecionados alguns cenários para simulação; na sequência, os resultados da execução dessas simulações foram validados estatisticamente e, por fim, as diversas variações dos parâmetros permitiram obter análises descritivas a respeito dos diferentes cenários gerados.

No capítulo 7, são apresentadas as conclusões deste trabalho de pesquisa e as propostas para trabalhos futuros.

2. TRABALHOS RELACIONADOS

Para a execução deste trabalho foi consultada a bibliografia em periódicos recentes e revistas indexadas nas principais bases de dados *online* disponíveis para busca. Com isso, pôde-se montar um panorama geral do estado da arte relacionado com esta pesquisa, mostrando os principais tópicos pesquisados e oportunidades de avanço do conhecimento. Busca-se, assim, verificar como este trabalho se insere no contexto dos demais trabalhos sendo feitos na área.

Assim, pôde-se verificar se os trabalhos relacionados ao tema de pesquisa utilizam Sistemas Múltiplos Agentes (MAS – *Multiple Agent System*), lidam com múltiplas aeronaves, se a trajetória da aeronave considerada é referencial a um ambiente 2D ou 3D, etc. (Beard, 2012).

Têm-se, também, questões a respeito do foco da pesquisa, ou seja, se elas consistem na criação de rotas (R) ou trajetórias (T). Rotas são estáticas e não consideram o deslocamento no tempo, diferentemente das trajetórias, que consideram a velocidade e movimentação das aeronaves. Define-se uma rota como uma série de pontos que levam um veículo de um ponto até outro ponto, podendo-se considerar, por exemplo, nesse processo, a presença de obstáculos. A rota pode ser calculada antes de o veículo partir, considerando um mapa estático do ambiente a ser percorrido. Por outro lado, a trajetória do veículo considera o tempo, ou seja, ela é dinâmica no sentido em que leva em conta fatores como a velocidade do veículo e as mudanças que devam ser feitas na construção dessa trajetória para evitar colisões; dessa forma, a trajetória corresponde ao caminho que o veículo efetivamente tenta seguir durante sua viagem entre seu ponto de partida e o seu destino (Beard, 2012).

Além disso, nessa pesquisa bibliográfica, desconsideram-se trabalhos que focam apenas em *path following* (“seguir o caminho”), que são necessários na implementação real, mas não são o foco principal deste trabalho, que se situa na camada superior de planejamento de trajetórias (Beard, 2012).

Em Lei (2011) é descrito um exemplo de um sistema multiagente para controlar uma única aeronave em um campo inimigo, na presença de diversos radares inimigos, utilizando uma técnica inteligente evolucionária conhecida como “aprendizagem por reforço”. O objetivo é gerar uma rota ótima (mais curta) até o destino, que necessariamente não passe nas áreas de radar.

Destaca-se em Besada-Portas et al. (2013) um trabalho que busca criar rotas para uma aeronave com o objetivo de evitar obstáculos (prédios). Este trabalho não prioriza a segurança, já que considera que podem existir colisões, apesar de tentar minimizá-las. Tenta-se, finalmente, encontrar um método inteligente evolutivo que gere menos colisões e, principalmente, reduza o caminho percorrido pela aeronave. Para isso, são comparados 36 algoritmos diferentes.

No artigo de Chen (2016) trata-se do uso de 3 aeronaves em um projeto colaborativo de construção, no qual as aeronaves dividem tarefas em um mesmo projeto de construção, buscando e posicionando materiais de acordo com um projeto de construção compartilhado entre os agentes e, assim, precisam dividir o espaço aéreo. O trabalho visa otimizar de maneira segura e colaborativa essas trajetórias, de forma dinâmica, considerando um ambiente em 3D (três dimensões), com o uso de algoritmos genéticos.

No artigo de Wei (2015) os autores descrevem um sistema que visa auxiliar o Controlador de Tráfego Aéreo (ATCo – *Air Traffic Controller*) na geração de trajetórias ótimas para aeronaves diversas dividindo o mesmo espaço aéreo. Essa abordagem utiliza-se de sistemas multiagentes e lida com múltiplas aeronaves.

Em Yang, Fang e Lee (2016) tem-se uma proposta utilizando um método chamado de “Teoria Tau”, que é utilizado para executar a otimização de uma equação objetivo utilizando-se do método conhecido como PSO (*Particle Swarm Optimization*). Com o uso do PSO, pretende-se reduzir as trajetórias de maneira genérica, gerando estratégias ótimas ou próximas da ótima que sejam seguras (nas quais não haja colisões), dentro de um ambiente em 3D.

Em Coelho et al. (2017) apresenta-se um trabalho de otimização de trajetórias de múltiplas aeronaves utilizando-se, para isso, de um ambiente 3D. É empregada, nessa pesquisa, uma técnica conhecida como MILP (*Mixed-Integer Linear Programming*) para gerar a otimização, com foco na redução do uso de energia (bateria) dos *drones*, promovendo, assim, a redução de custos e, de certa forma, uma maior preservação da natureza.

Em Wen (2017) é utilizado um método chamado de RRT (*Rapidly-exploring Random Tree*), em um ambiente 3D, para gerar a rota de uma única aeronave em um ambiente hostil.

No trabalho de Silva (2018) é descrito um método para gerar rotas para múltiplas aeronaves, que possuem um controle centralizado, e cujo objetivo é propiciar

com que essas aeronaves executem uma série de tarefas com a menor trajetória total, considerando todas essas aeronaves, que passariam a ter, assim, caminhos menores a percorrer. Para isso é feita uma divisão de tarefas e cálculo de rotas usando métodos matemáticos de otimização não-linear para a obtenção de um resultado ótimo.

Em Bo et al. (2017) é usado o algoritmo A* para criar trajetórias ótimas de uma aeronave em um campo com obstáculos. Esse algoritmo gera uma série de pontos fixos e a aeronave tenta manter sua trajetória próxima a esse caminho ideal inicial. O autor pretende usar essa abordagem para criar rotas com múltiplas aeronaves.

Em Zhao et al. (2018) busca-se gerar trajetórias ótimas de uma aeronave navegando em um ambiente 3D. Essas trajetórias são otimizadas com base somente em informações limitadas coletadas pelos sensores da aeronave.

Apresentam-se, a seguir, na Tabela 1, os aspectos mais relevantes de cada uma das pesquisas citadas, para a identificação das oportunidades de pesquisa a serem percorridas por este trabalho

Tabela 1 – Resumo: Trabalhos Relacionados

	Trabalho Considerado	MAS	Múltiplas Aeronaves	Tipo Ambiente	Rota/ Trajetória	UAV	Objetivo	Detalhe
1	(Lei, 2011)	Sim	Não	2D	Rota	Sim	Desvio de Obstáculos	Aprendizagem por reforço
2	(Besada-Portas et al., 2013)	Não	Não	2D	Rota	Sim	Evitar Obstáculos	36 métodos evolucionários diferentes
3	(CHEN et al., 2016)	Não	Sim	3D	Trajetoária	Sim	Construção colaborativa de Trajetórias	Algoritmos Genéticos
4	(WEI, 2015)	Sim	Sim	3D	Trajetoária	Não	Sistema visa ajudar o ATCo (não é específico para veículos autônomos)	-
5	(YANG; FANG; LI, 2016)	Não	Sim	3D	Trajetoária	Sim	Prevenção de colisões	PSO
6	(COELHO et al., 2017)	Não	Sim	3D	Trajetoária	Sim	Economia de energia	MILP

	Trabalho Considerado	MAS	Múltiplas Aeronaves	Tipo Ambiente	Rota/Trajatória	UAV	Objetivo	Detalhe
7	(WEN et al., 2017)	Não	Não	3D	Rota	Sim	Rotas em Ambiente hostil	RTT
8	(Silva; Rajasinghege, 2018)	Não	Sim	3D	Rota	Sim	Controle centralizado	Programação Não-Linear
9	(Bo et al., 2017)	Não	Não	2D	Trajatória	Sim	-	A*
10	(Zhao et al., 2018)	Não	Não	3D	Trajatória	Sim	Simulação de sensores	-

Fonte: Autor. (2022)

A Figura 1 apresenta um resumo das principais diferenças entre os objetivos deste trabalho e os investigados no estado-da-arte da bibliografia, conforme o que foi apresentado na Tabela 1. Destaca-se em verde os itens que possuem uma relação direta com este trabalho de pesquisa, enquanto que na cor vermelha indica-se os pontos divergentes com relação ao que está sendo desenvolvido neste trabalho.

Como pode-se observar nesses diversos trabalhos de pesquisa, existe um tema relevante (está sendo trabalhado por diversos autores), apesar de múltiplas abordagens parecidas nenhuma dessas abordagens contém todas as características destacadas neste trabalho, mostrando a contribuição específica desse trabalho nesse nicho específico.

	MAS	Múltiplas Aeronaves	Tipo Ambiente	R/T	UAV
1	Verde	Vermelho	Vermelho	Vermelho	Verde
2	Vermelho	Vermelho	Vermelho	Vermelho	Verde
3	Vermelho	Verde	Verde	Verde	Verde
4	Verde	Verde	Verde	Verde	Vermelho
5	Vermelho	Verde	Verde	Verde	Verde
6	Vermelho	Verde	Verde	Verde	Verde
7	Vermelho	Vermelho	Verde	Vermelho	Verde
8	Vermelho	Verde	Verde	Vermelho	Verde
9	Vermelho	Vermelho	Vermelho	Verde	Verde
10	Vermelho	Vermelho	Verde	Verde	Verde

Figura 1 – Resumo estado da arte da pesquisa em simulação do UAM com MAS

Fonte: Autor (2022)

3. SISTEMAS MULTIAGENTES

Neste capítulo são introduzidos alguns temas importantes na área de computação utilizados nesse trabalho. Inicialmente apresenta-se aspectos teóricos relacionados com o paradigma baseado em agentes e a sua importância na área de Inteligência Artificial. Posteriormente, os conceitos relacionados com sistemas multiagentes são introduzidos considerando a linguagem Logo e o Netlogo.

3.1 AGENTES

Um agente, em sua definição mais simples, é um indivíduo (ou instrumento) capaz de agir ou que tem o poder de agir. Pode-se definir um agente como um indivíduo que tem uma capacidade de visualizar um ambiente (mundo) e interagir com esse ambiente, potencialmente alterando o ambiente com suas ações. O conceito de seres que podem ser agentes remonta há muito tempo, percorrendo o contexto de seres autônomos e de máquinas antigas, estando conceitualmente presente desde o início da computação e do campo da Inteligência Artificial (IA) na década de 1950. Sendo assim, o paradigma de agentes na construção de programas pode ser visto como um método eficiente de abstração no desenvolvimento de sistemas inteligentes (Bradshaw, 1997).

Muitos autores, como Russell e Norvig (2009), dentre outros, têm usado o paradigma orientado a agentes como principal forma de desenvolvimento de sistemas que envolvem o uso de IA, tanto na academia quanto na indústria. Isso se deve à sua simplicidade e à facilidade de abstração, podendo hoje ser considerado o principal método utilizado na área de inteligência artificial para trabalhar com a abstração necessária em sistemas complexos.

Neste contexto, existe na literatura toda uma área conhecida como Modelagem Baseada em Agentes (ABM – *Agent Based Modelling*). Essa modelagem é baseada na representação de problemas das mais diversas áreas (modelos sociais, físicos e biológicos) em uma perspectiva de agentes, de forma que tais problemas possam ser mais facilmente entendidos e resolvidos por meio de modelos (Macal e North, 2010).

Existem, inclusive, metodologias de desenvolvimento de ABM visando facilitar e dividir esse desenvolvimento em etapas, buscando-se obter um *design* (projeto)

mais simples e de maneira mais estruturada e que possa ser utilizado em projetos de desenvolvimento de grande porte na indústria. Em (Macal e Nort, 2010) descreve-se um exemplo da metodologia mais utilizada que consiste em um método iterativo, composto por 6 passos, e que responde a uma pergunta por passo. As questões são:

- 1) O que o modelo procura responder? Ou qual problema precisa ser respondido? (Problema).
- 2) Quais agentes vão existir no modelo (e quais não vão)? Quais dados eles terão e seus comportamentos (Agentes).
- 3) Qual é o ambiente dos agentes? Como os agentes interagem e quais características esse ambiente possui (Ambiente).
- 4) Quais comportamentos dos agentes de interesse para resolução do problema? (Comportamento).
- 5) Qual a relação do agente com os outros agentes e o ambiente? (Relações).
- 6) Quais dados estão sendo usados? Ou quais dados estão sendo gerados (e validados)? (Modelo).

3.2 MULTIAGENTES

Sistemas Multiagentes (*Multiagent Systems* – MAS) constituem uma subárea que une IA (Inteligência artificial) e sistemas distribuídos, conhecida como Inteligência Artificial Distribuída (DAI – *Distributed Artificial Intelligence*), cujo objetivo principal é verificar o desempenho conjunto e comunicação entre esses agentes, permitindo o aprendizado e interação em situações de ambientes complexos (Panait, 2015).

MAS são sistemas distribuídos e interconectados, compostos de múltiplas entidades autônomas chamadas de agentes (ou seja, são sistemas modelados no paradigma dos agentes e são compostos por mais de um agente). Sistemas Multiagentes são muito úteis pois permitem a colaboração entre diversos sistemas diferentes e são capazes de criar sistemas que apresentam alta confiabilidade e tolerância a falhas, sendo utilizados em sistemas críticos como os de telefonia e de aviação (Wooldridge, 2002).

Nos Sistemas Multiagentes, cada agente tem a capacidade de interagir com o ambiente e, assim, coletar informações, executar determinadas ações no ambiente e trocar mensagens com outros agentes. Alguns pontos principais dos agentes que eles devem possuir como principais características são: autonomia (agem de maneira

independente dos outros agentes), proatividade (tomam iniciativa), reatividade (são capazes de perceber mudanças no ambiente e nas situações, bem como alterar seu comportamento) e habilidade social (são capazes de se comunicar com outros agentes e, assim, obter informações e ajuda) (Wooldridge, 2000).

Um outro paradigma colocado nos sistemas multiagentes é o chamado BDI (*Belief–Desire–Intention*), ou crença-desejo-intenção. Nesse paradigma, pode-se representar um agente pelas suas crenças (ou conhecimentos), pelos seus desejos (ou objetivos) e pelas suas intenções (ou ações que ele pode tomar) (Wooldridge, 2002).

Além disso, outros autores colocam também como características dos Sistemas Multiagentes a continuidade temporal (agente mantém seu *status* e conhecimento ao longo do tempo), personalidade (o agente desenvolve características pessoais como confiabilidade perante outros e emoções), adaptabilidade (o agente desenvolve a capacidade de se adaptar a diferentes situações) e mobilidade (agente desenvolve a capacidade de se mover em diferentes cenários e ambientes) (Bradshaw, 1997).

Deve-se destacar que muito se confunde a eventual similaridade entre o paradigma orientado a agentes e o paradigma da orientação a objetos. Em certas situações acredita-se que ambos possuem certas semelhanças ou sejam até mesmo equivalentes. Porém, na orientação a objetos existe a comunicação entre objetos, mas neste caso os objetos trocam mensagens e simplesmente obedecem ao que lhes é pedido; assim, os objetos não possuem individualidade e independência, são simples componentes de um sistema maior, ao contrário do que ocorre com um agente em um Sistema Multiagente (Wooldridge, 2002).

Existem agentes com graus de autonomia variados. Uma classificação simples pode ser feita entre agentes reativos e agentes cognitivos. Os agentes reativos são o tipo de agente mais simples possível, pois eles agem de maneira autônoma em relação aos outros agentes, tomando decisões de acordo com que eles podem “sentir” (informações coletadas) do ambiente, e reagem de maneira determinada a essas situações; suas relações com outros agentes que estão dividindo o mesmo ambiente se resumem a essa mesmas ações determinadas, não havendo troca de informações; porém, apesar disso, esses agentes têm capacidade de resolver de maneira simples, mas não necessariamente ótima, muitos problemas. Já os agentes cognitivos são agentes que possuem interação com outros agentes, coletando e trocando

informações e realizando negociações; com isso, esse tipo de agente constrói uma base de conhecimento muito mais complexa para ser usada em tomadas de decisão (Wooldridge, 2000).

Quanto à sua autonomia, pode-se dividir os agentes naqueles que estão dispostos a cumprir efetivamente sua tarefa (que pode levar ao caso de problemas em que o objetivo não seja mais possível de ser alcançado) ou aqueles que são mais flexíveis quanto aos objetivos passados. Normalmente deve-se ajustar o nível de autonomia do agente de acordo com a tarefa que este deve realizar, sendo como muitos parâmetros existentes dos agentes necessita ser alterado de acordo com o cenário (muitas vezes durante a própria execução) (Macal e Nort, 2010).

Outra questão importante a ser vista na abordagem dos Sistemas Multiagentes é a comunicação entre os agentes. Durante a modelagem, os agentes podem se comunicar com todos os demais agentes ou apenas com grupos de agentes. Os agentes podem ter a capacidade de negociação, de estarem formados em times ou serem capazes de se organizar em times de forma a gerenciar seus recursos, podendo executar leilões para dividir suas tarefas e ter confiança de outros agentes (podendo firmar parcerias entre agentes e, dependendo da configuração, violar essas parcerias ou prejudicar agentes, agindo fora das regras definidas nesta sociedade de agentes) (Wooldridge, 2002).

Os agentes possuem uma configuração interna que não é acessível por outros agentes, na qual ele armazena sua base de conhecimento e, por exemplo, guarda os conhecimentos necessários sobre o ambiente e outros agentes, e onde também guarda informações sobre seus objetivos. Assim os agentes possuem estados no mundo (variáveis que dizem respeito a relação entre o agente e o ambiente, exemplo localização física e relacionamentos entre agentes) e memória. Esta configuração também define quais ações um agente pode executar e como ele se comunica com sua exterioridade (Macal e Nort2010).

Existe, também, outro fator a ser considerado durante a modelagem de um Sistema Multiagente: a topologia do ambiente. Os agentes podem interagir não só entre si, mas principalmente com o ambiente. A definição de como vai ser o ambiente diz muito sobre como o sistema irá funcionar. Pode-se exemplificar isso com o caso de agentes estarem trabalhando em espaços com uma topologia em forma de ambiente 2D ou 3D, no espaço euclidiano ou discreto (autômatos celulares). Também existem topologias em formas de rede ou modelos a-espaciais (Macal e Nort, 2010).

Nos modelos a-espaciais, a relação entre os agentes é importante, mas eles não possuem uma localização no sentido espacial. Por exemplo, considerando um grupo de agentes que têm que resolver uma equação, a sua topologia vai se referir a como eles se relacionam e qual a divisão de tarefas, sendo que essa relação não tem nada associado a um espaço físico, já que esses agentes não estão colocados em uma área do espaço (Macal e Nort, 2010).

A forma como a divisão do ambiente pode ser feita leva a que nem todos os agentes precisam estar no mesmo ambiente; além da forma e tamanho deste ambiente serem variáveis, o modelo pode fazer a combinação de diversos tipos de topologias mais simples em formas mais complexas (Macal e Nort, 2010).

Pode-se perceber, assim, que a fase de modelagem (construção do modelo) em que é feita a abstração da situação real em um modelo é uma fase complexa, e existem muitas variáveis e variações possíveis em MAS, tanto dos agentes quanto do ambiente, de forma a gerar uma simulação mais precisa do problema que está sendo tratado.

Os exemplos mais bem-sucedidos do uso de Sistemas Multiagentes mostrados na literatura concentram-se, principalmente, em sistemas complexos, sistemas críticos e sistemas distribuídos. Existem alguns exemplos de sucesso de Sistemas Multiagentes implantados na indústria e bem documentados como, por exemplo, sistemas de telefonia e sistemas de aviação (especificamente com um sistema para auxiliar o controle de tráfego aéreo) (Wooldridge, 2002).

Os Sistemas Multiagentes são sistemas conhecidos por sua tolerância a falhas e escalabilidade, além de sua implementação em vários tipos de aplicações em diferentes áreas. Além disso, existem *frameworks* em diversas plataformas e linguagens de programação. Muitas dessas plataformas são *open-source* (possuem código fonte aberto e atualizado por comunidades acadêmicas e de usuários) (Xie; Liu, 2017).

Como é possível observar na Figura 2, há um agente que é uma entidade fechada em si, isolada do ambiente, possuindo algumas características internas como uma **Base de Conhecimento**, que pode ser constituída de informações, como conhecimento sobre o mundo. O agente apesar de ser fechado em si também tem a capacidade de enxergar o **Estado do Ambiente** e, assim, conhecer se existem agentes próximos ou se existe alguma condição específica no ambiente que o cerca. E, por último, o agente tem também uma série de **Habilidades** (ou **Funções**) nas

quais ele pode interagir com o ambiente, podendo mover-se, escanear o ambiente e coletar algum objeto.



Figura 2 – Arquitetura de um Agente

Fonte: Autor (2022). Adaptado de Wooldridge (2002) e Macal e Nort (2010)

Na Figura 3, apresenta-se um diagrama exemplificando de como é um sistema multiagente. Por meio de uma visão de cima, existem múltiplos agentes, sendo que cada agente individual possui diversas características internas, como as mostradas anteriormente. Um agente, então, interage, enviando e recebendo *feedback* tanto do ambiente no qual ele está inserido quanto de todos os demais agentes, atualizando seus estados internos e afetando os estados internos dos outros agentes.

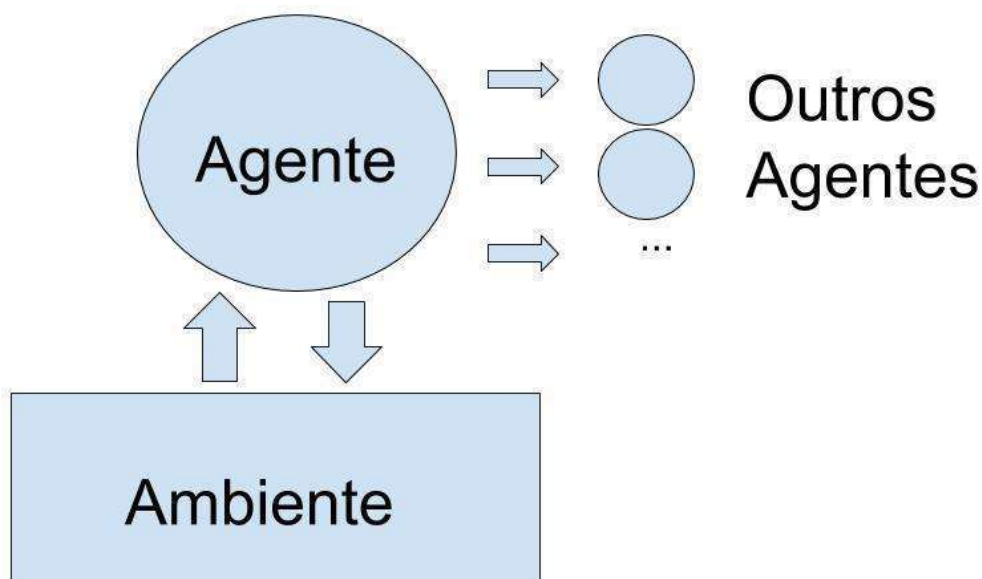


Figura 3 – Arquitetura de um MAS

Fonte: Autor (2022). Adaptado de Wooldridge (2002) e Macal e Nort (2010)

3.3 LINGUAGEM NETLOGO

A Linguagem Logo foi criada no final da década de 1960 e fez bastante sucesso durante a década de 1970. Inicialmente foi desenvolvida nos laboratórios do MIT (*Massachusetts Institute of Technology*) pelos cientistas da computação Seymour Papert, Wallace Feurzeig e Cynthia Solomon como uma linguagem simples e introdutória, fortemente baseada na linguagem LISP (linguagem que, por sua vez, era fortemente voltada para a Inteligência Artificial – IA), focada para o ensino de matemática, lógica e programação para crianças (Papert, 1980).

Papert, além de pesquisador na área de IA, era interessado pela área de educação de crianças pequenas. Baseado na pedagogia desenvolvida pelo pedagogo suíço Jean Piaget, que desenvolveu alguns anos antes a pedagogia do construtivismo, Papert, baseado e influenciado nas ideias de Piaget e em suas próprias experiências desenvolveu uma pedagogia derivada chamada de construcionismo, que foi aplicado no desenvolvimento do Logo (Papert, 1980).

Sendo o Logo uma linguagem orientada a um agente (no singular), ela se concentra em gerar comandos para um robô (agente) na forma de uma tartaruga, que na década de 60 seria um robô capaz de desenhar no chão, e que com o desenvolvimento da tecnologia dos monitores, evoluiu para uma versão sua digital.

Além disso, a linguagem Logo, assim como a linguagem LISP, encontrou aceitação em aplicações mais complexas, incluindo a Inteligência Artificial (Papert, 1980).

Na década de 1990, o cientista da computação Uri Wilesky, trabalhando com diferentes implementações da linguagem Logo, criadas durante os anos, resolveu iniciar o desenvolvimento de uma própria implementação do Logo, chamado de Netlogo. Sua ideia seria criar uma extensão da linguagem Logo para poder suportar o uso de sistemas multiagentes e, assim, propiciar a criação e controle de múltiplos agentes, de forma a estender a linguagem já existente e testada do Logo (Wilesky, 1999) (Wilesky, 2015).

Tanto a linguagem do Netlogo quanto o seu código fonte foram disponibilizados de forma aberta para serem melhorados e estendidos por uma ampla comunidade *online* (Wilesky, 1999). O Netlogo possui duas versões: uma versão primariamente 2D, mas parecida com o Logo; e uma extensão da versão 2D em sua versão 3D, que permite não só que seus agentes se desloquem em um ambiente 2D, como também em um ambiente 3D (Wilesky, 2015).

A Linguagem Netlogo pode ser aprendida de maneira simples e é muito poderosa, podendo ser utilizado de maneira versátil para a construção de simulações e modelos, que podem ser simples ou extremamente complexos. Tais modelos podem ser utilizados na construção de diversos tipos de cenários complexos, para utilização nas áreas de ciências sociais, engenharia, física e ciência da computação (Wilesky, 2015).

O ambiente de desenvolvimento do Netlogo (IDE Netlogo) também facilita o desenvolvimento, depuração e visualização da execução do código desenvolvido (Wilesky, 2015). O Netlogo também conta com uma extensiva biblioteca de modelos, com exemplos nas mais diversas áreas (aviação, física, química, ciências sociais, etc.), disponibilizando manuais e guias de referência *online* para que tais exemplos possam ser compreendidos e utilizados por pesquisadores das mais diversas áreas (Wilesky, 2015). Pode-se citar, em particular, o exemplo do modelo de uma aeronave de asa fixa pousando em uma pista, como um exemplo para uso em simulação de aeronaves (Wilesky, 2011).

4. UNMANNED AIRCRAFT SYSTEMS (UAS)

Neste capítulo são apresentados alguns componentes e tecnologias que compõem os sistemas utilizados em aeronaves autônomas, utilizadas no desenvolvimento deste trabalho de pesquisa. No primeiro tópico são introduzidos conceitos relacionados com aeronaves autônomas e diferentes terminologias empregadas. Já no segundo tópico descreve-se o funcionamento do sistema de controle de tráfego aéreo tradicional. Por fim, no terceiro e último tópico são abordados os sistemas do tráfego aéreo aplicados nas aeronaves autônomas.

4.1 UNMANNED AERIAL VEHICLES (UAVs)

UAVs (*Unmanned Aerial Vehicles*) ou *drones* são aeronaves capazes de voo autônomo ou semiautônomo (como as aeronaves controladas remotamente). Apesar de não ser algo novo, trata-se de uma tecnologia em desenvolvimento e que está recebendo muita atenção nos últimos anos. Os UAVs são muito úteis em aplicações de risco (como nas tarefas de grandes construções, inspeção de obras e resgate de pessoas em situação de risco) e em casos em que se queira construir aeronaves mais leves (seja para o transporte de carga ou até mesmo de passageiros) (Shakhatreh et al., 2018).

UAS (*Unmanned Aircraft Systems*) são sistemas compostos pelos UAVs (considerando os seus sensores e os seus sistemas embarcados) e pelos Sistema de missão (como são chamados os sistemas em solo, que controlam a missão da aeronave), que englobam sistemas de controle remoto e integração com ATC (*Air Traffic Control*) (Shakhatreh et al., 2018).

Os UAVs correspondem a uma grande quantidade e variedade de aeronaves. Basicamente, qualquer aeronave tradicional pode ser autônoma e muitos tipos de aeronaves estão sendo usados para aplicações envolvendo *drones*. Particularmente, existem algumas classes de aeronaves do tipo *drones*, quem podem ser classificadas pela forma como executam a decolagem: os veículos de decolagem horizontal (CTOL – *Conventional Take-Off and Landing*), que são a classe de veículos aéreos mais comuns, e os veículos de decolagem vertical (VTOL – *Vertical Take-Off and Landing*), cujos exemplos mais comuns são os helicópteros e os quadricópteros (ANAC, 2017).

Na categoria dos veículos do tipo VTOL encontram-se o eVTOL (*electric Vertical Take-Off and Landing*), que segue o mesmo conceito do VTOL, mas que possui grande potencial para o seu uso no ambiente urbano, já que não só eles têm a facilidade de decolagem e pouso na vertical, mas também possuem motores elétricos, além de uma ou múltiplas baterias, que podem ser carregadas com relativa facilidade. Além disso, comparativamente com veículos similares, o eVTOL é mais leve e tem uma emissão de poluição mínima, podendo utilizar, para o seu carregamento, fontes de energia renováveis. Dessa forma, a autonomia do eVTOL é limitada apenas pela qualidade das baterias atualmente existentes, sendo que o desenvolvimento tecnológico nessa área está avançando de forma acelerada (Jain et al., 2020).

Os UAVs podem ter diferentes níveis de autonomia, existindo aqueles capazes de voo totalmente autônomo, e os de voo apenas parcialmente autônomo. Apesar dos avanços e aumento de autonomia de voo dos UAVs, ainda não existem aeronaves totalmente autônomas. Um ponto importante a ser considerado é que, mesmo levando-se em conta um veículo como autônomo, sua missão será sempre determinada por um indivíduo ou *software* externo. Um tipo muito conhecido de aeronave parcialmente autônoma é o RPAS (*Remotely Piloted Aircraft Systems*), que são as aeronaves pilotadas remotamente, diferentemente das autônomas, que são pilotadas totalmente por um sistema embarcado. O RPAS possui um piloto humano tomando decisões e controlando a aeronave via rádio, mas, apesar disso, essas aeronaves possuem um alto grau de autonomia, maior do que o das aeronaves convencionais, já que o sistema pode ter que tomar decisões autônomas caso a conexão via rádio seja interrompida (ANAC, 2017).

Os RPAS, dependendo do número de sensores presentes nas aeronaves e do nível de automação implementado, podem ser divididos em subtipos: VLOS (*Visual Line of Sight*), EVLOS (*Extended Visual Line of Sight*) e BVLOS (*Beyond Visual Line of Sight*). Esses tipos são abordados na sequência deste capítulo.

Para controlar as trajetórias dos UAVs, não poderia ser diferente. Em Beard (Beard, 2012) propõe-se uma arquitetura comum aos sistemas que controlam a movimentação (trajetórias) dessas aeronaves. Na Figura 4 pode-se observar como o UAV recebe informações, no nível de *software*, de outras aeronaves e do ATM, e as informações do ambiente, no nível de *hardware*, por meio de sensores.

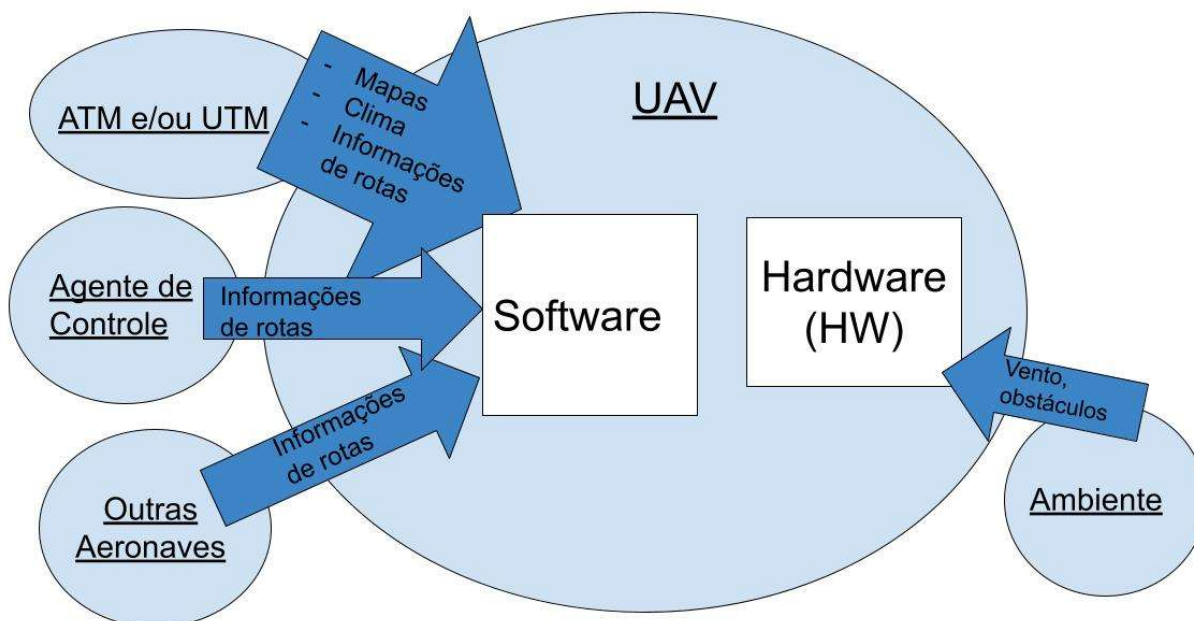


Figura 4 – Arquitetura do UAS: entrada de dados externos

Fonte: Autor (2022). Adaptado de Beard (2012).

Com isso, pode-se separar os subsistemas do UAV em camadas, conforme a Figura 5, para melhor compreensão, e com foco na resolução de problemas específicos relacionados com cada nível. A primeira camada é o *Autopilot*, que é responsável por fazer ajustes e controlar o *hardware* da aeronave; a segunda camada é o *Path Following*, que foca em corrigir desvios da aeronave do caminho ideal; a terceira camada é o *Path Manager*, que define, a partir do caminho planejado, o caminho ideal a ser seguido, definindo, assim, a velocidade necessária e quais manobras precisam ser realizadas; e a quarta camada é o *Path Planner*, que informa qual caminho deve ser seguido pela aeronave (Beard, 2012).

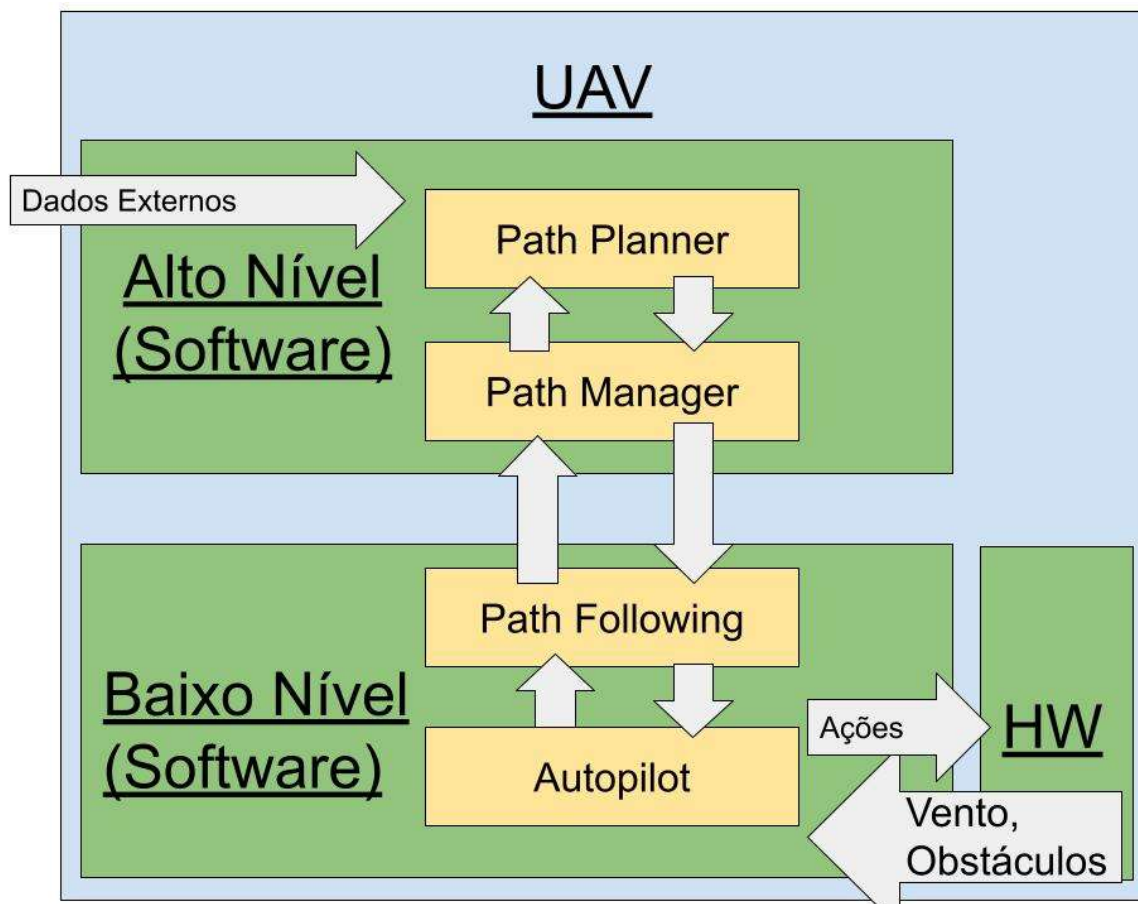


Figura 5 – Arquitetura do UAS: controle de caminho

Fonte: Autor (2022). Adaptado de Beard (2012).

Ainda na Figura 5, tem-se uma descrição da arquitetura do *software* dentro de um UAV. Ele recebe dados externos dentro da camada de *Path Planer* que está no mais alto nível na arquitetura, sendo que a camada do *Autopilot* lida com a parte de *hardware* da aeronave.

Na arquitetura de alto nível da Figura 5, existem dois níveis alimentando o sistema de *software* e um de *hardware* em que efetivamente existem os atuadores para mudar a direção e posição da aeronave. Além disso, o nível de *hardware* também recebe informações dos sensores da aeronave.

A aeronave pode ter um receptor interno que permite, no alto nível, o recebimento de informações desde mapas, rotas, clima, áreas restritas, até troca de informações entre outras aeronaves e entidades (chamado de *datalink* ou *link* de informações) (Beard, 2012). No baixo nível, o módulo do *Autopilot* cuida de fazer a parte mais baixa de mudanças e correções e segurança de desvio de obstáculos imediatos. Já a parte de "*Path Following*" tem, em baixo nível, função de verificar o

desvio de rotas da aeronave e fazer acertos simples de caminho em tempo real. No nível mais alto, o *Path Planner* realiza um nível mais alto de planejamento, em que principalmente existe uma definição de rotas, sendo que o *Path Manager* tem a função de selecionar as rotas de acordo com as informações recebidas. Portanto, existem alguns trabalhos com um foco maior no “nível baixo” ou no “nível alto” dessa arquitetura.

Em particular, este trabalho foca na parte de construção de modelos de definição de rotas e seu gerenciamento. Assume-se que o “nível mais baixo” é minimamente capaz de fazer as correções necessárias e operação de maneira eficiente de forma a não interferir no nível estudado nesta pesquisa.

Nesse sentido, pode-se ainda apontar que as camadas superiores de *Path Planner* e *Path Manager* estão mais ligadas a aspectos computacionais de geração de caminhos e gerenciamento de obstáculos, enquanto que as camadas de *Path Following* e *Autopilot*, além de receberem informações dos níveis superiores, trabalham com correções de mais baixo nível na operação do UAV, como interagindo com sensores de visão (para efetuar desvio de pássaros e de outros obstáculos, por exemplo) e correções de rotas (devido a ventos).

Existem vários tipos de veículos aéreos. Um tipo específico de veículo aéreo são os helicópteros, que são uma classe de veículos movidos a hélices rotativas do tipo VTOL. Os helicópteros tradicionais se movem com duas hélices, uma na horizontal, a asa rotativa, e outra na vertical, o rotor de cauda (que serve para dar estabilidade), chamados de veículos mono-rotores. Por outro lado, existem helicópteros multi-rotores, que são todos os tipos de veículos que possuem duas ou mais hélices verticais. Os veículos multi-rotores são conhecidos pela sua capacidade de manobra e adaptabilidade em várias situações, e utilizados principalmente na construção de *drones*, como os quadricópteros (quatro rotores) e os octocópteros (oito rotores), sendo os quadricópteros os mais populares. Pode existir, basicamente, qualquer configuração de número de motores, devendo apenas existir um número par deles (pois um gira em sentido contrário ao seu par, gerando a estabilidade da aeronave) (Kim, 2018). Os quadricópteros (e outros multi-rotores) são conhecidos por serem veículos simples e com grande capacidade de voo, possuindo seis graus de liberdade em manobras (podem manobrar em qualquer direção), com uso de pouco espaço de manobra, com capacidade de decolar e pousar em ambientes pequenos (decolagem do tipo VTOL), com capacidade de ficar “parado no ar” (voo pairado), etc.

Essas características tornam esse tipo de aeronave ideal no caso de uso de UAVs em centros urbanos (principalmente pelas manobras entre prédios), aplicações de filmagem e de *drones* capazes de executar tarefas específicas com precisão (Beard, 2012).

4.2 AIR TRAFFIC MANAGEMENT (ATM)

O ATC (*Air Traffic Control*) corresponde aos sistemas e pessoas que controlam o uso do espaço aéreo, tendo com tarefas assegurar as separações entre aeronaves e o uso racional do espaço aéreo pelas aeronaves, para garantir a segurança operacional, bem como o uso justo desse espaço pelos diversos participantes (Shakhatreh et al., 2018). O ATCo (*Air Traffic Control Operator*) é o termo usado para se referir aos controladores de tráfego aéreo. O ATC inclui todo o controle de trajetórias e permissões das aeronaves em voo. Quando as aeronaves entram em fase final de voo, elas são transferidas para o AGC (*Air Ground Controller*) que é o responsável pelo pouso seguro da aeronave no aeroporto. O *Air Traffic Management* (ATM) é o sistema de controle de tráfego aéreo que inclui o ATC, o ATCo, os pilotos e os sistemas utilizados no controle de tráfego aéreo (Lundberg et al., 2018).

Atualmente, não existe um sistema de ATM para gerenciar os voos dos UAVs. Desta forma, os UAVs voando em espaço aéreo não-segregado seguem as mesmas normas das agências reguladoras internacionais, como a FAA (*Federal Aviation Administration*) nos Estados Unidos e a EASA (*European Aviation Safety Agency*) na Europa (ANAC, 2017). Isso inclui a utilização dos padrões de separação pertinentes e o uso do ATC, o que não se mostra prático e viável no cenário futuro, pois isso resultaria em um espaço aéreo com capacidade muito reduzida, já que as aeronaves maiores possuem parâmetros de segurança mais conservadores devido ao seu tamanho e ao controle manual da aeronave, apesar da intensa automação no decorrer dos anos que permitiu a redução das distâncias entre aeronaves, mantendo o mesmo nível de segurança. Tudo isso teria impacto na eficiência do sistema. Além disso, aeronaves de asa fixa tem por característica deixar um “rastro de turbulência” no ar, o que aumenta os limites de separação horizontais entre aeronaves, principalmente nos quesitos de pousos e decolagens nas pistas dos aeroportos, algo que não faz sentido para as operações envolvendo VTOLs.

Assim, dentro deste contexto, têm-se discutido mundialmente como serão as normas de segurança e padrões futuros a respeito. No Brasil, isso não tem sido diferente, e as agências locais nos diferentes países, como a ANAC (Agência Nacional de Aviação Civil), no Brasil, têm se esforçado na criação e definição de padrões internacionais a serem seguidos, de maneira a manter a segurança e a eficiência no uso do espaço aéreo nacional (ANAC, 2017).

Adicionalmente, têm-se discutido a criação de regras de voo autônomas (*Autonomous Flight Rules*) específicas para o gerenciamento de aeronaves autônomas (Wing, 2011). Mas, atualmente, as regras para esse fim cobrem somente ambientes segregados e são muito semelhantes às utilizadas para a aviação convencional (ANAC, 2017).

Apesar disso, a regulamentação atual para o uso de UAVs ainda é deficitária e carece de tecnologia e de eficiência. Basicamente, somente o uso do RPAS está regulamentado e a utilização de aeronaves totalmente autônomas ainda não possui uma regulamentação definida. Dessa forma, pelo menos teoricamente, aeronaves totalmente autônomas ainda não poderiam decolar e utilizar o espaço aéreo. Além disso, os RPAS devem voar em espaço aéreo segregado (não podem dividir o espaço aéreo com outras aeronaves convencionais) e a regulamentação atual não diferencia um VLLOS de um simples aeromodelo. Os RPAS têm a capacidade de voo reduzida e é necessário submeter toda a documentação de plano de voo preenchida de forma manual a um sistema, de maneira prévia, para que este tipo de aeronave possa executar a decolagem e realizar seu plano de voo, independente da tarefa ou missão que vai ser executada. Já para os sistemas que consideram o uso do BVLOS, que naturalmente tem uma autonomia maior, existem regulamentações de voo muito próximas às aplicadas às aeronaves convencionais (ANAC, 2017).

Ainda não existe na legislação uma caracterização específica para UAVs, e a maior parte da legislação apenas aproxima o uso de UAVs com o uso de outros tipos de aeronaves já existentes. Também não existe um consenso quanto à permissão de voos totalmente autônomos. Isso mostra que existe uma necessidade, não só tecnológica, como também conceitual para a resolução desse problema. Apesar deste trabalho de pesquisa não focar em uma realidade presente, ele pretende servir de base para esses desenvolvimentos futuros. Nesse trabalho, as aeronaves devem executar a missão de sair de um ponto A, onde estariam pousadas, e ir até um ponto

B (ambos os pontos dentro do espaço simulado), com o objetivo de efetuar uma entrega, seja de um passageiro, seja de um pacote.

Para o funcionamento do UAS são necessários diversos componentes e, quanto maior o nível de automação e independência de um operador humano, mais complexos e seguros devem ser esses sistemas. Como exemplo disso, tem-se a diferença entre aeronaves de voo visual (VLOS) e de voo por instrumentos (BVLOS). Esta última exige a introdução de diversos instrumentos (sensores) e respectivos sistemas. Desse modo, pode-se fazer com que o operador possa ter a capacidade de julgamento adequada ao executar os comandos e, além disso, possa detectar possíveis erros de comando, ou até mesmo que a aeronave possa tomar decisões autônomas (no caso em que o comando demore muito para chegar), evitando, assim, colisões e outros tipos de incidentes.

Além disso, quanto à separação, existem duas maneiras principais de separação de aeronaves no espaço aéreo. A primeira maneira é chamada de separação procedural, na qual o espaço aéreo é dividido em segmentos e, assim, dependendo da direção da aeronave, ela recebe um segmento de altitude específico, a separação pode ser feita por maneira visual (através de pontos de referência no solo) ou com uso de instrumentos como o uso de radar em solo que permite saber a partir de 4 pontos qual a direção a aeronave está em relação ao ponto. Isso, se por um lado tende a reduzir os conflitos, por outro lado também diminui a capacidade do espaço aéreo. A segunda maneira é o free flight, na qual permite-se o voo livre das aeronaves, não existindo a regra de separação entre elas, o que se por um lado pode aumentar os conflitos entre aeronaves navegando no espaço aéreo, por outro lado pode-se ganhar em capacidade do espaço aéreo (Wing, 2011).

Como pode ser observado na Figura 4, existem múltiplas divisões do espaço aéreo tradicional, sendo que as divisões são feitas de acordo com regiões de altitude e necessidades específicas ou condições de voo. O espaço aéreo é dividido em classes, A, B, C, D e E, usando diferentes regras, altitudes, velocidades estipuladas para as aeronaves e necessidades de uso de voo por instrumentos. Além dessas 5 divisões, existe a classe G, que é uma classe de espaço aéreo considerada “não controlada”, pois não possui um ATCo exercendo essa tarefa para essa classe do espaço aéreo. Mas mesmo assim, a região aglutinada pela classe G ainda possui regras básicas de voo visual para a navegação.

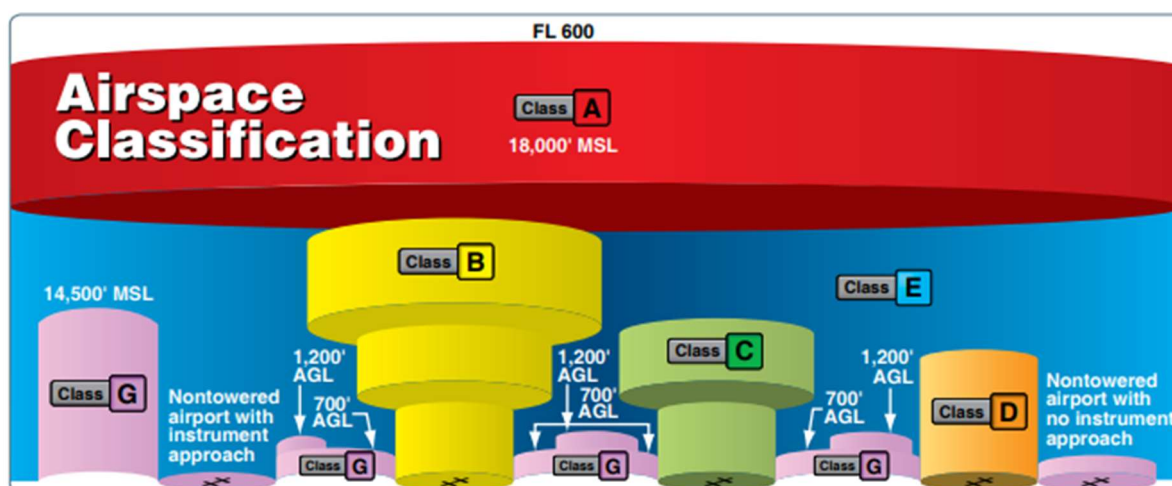


Figura 6 – Divisão do espaço aéreo tradicional em setores

Fonte: FAA (2016)

Existe a possibilidade também de serem considerados outros tipos de espaço, como áreas proibidas e restritas, e isso se deve ao fato de poderem existir algumas áreas com espaços sensíveis como aquelas no entorno de usinas nucleares, áreas de treinamento e bases militares, áreas de aeroportos civis, áreas de teste de aeronaves, entre outras, que exigem que o regulador coloque regras mais restritas nessas áreas ou até mesmo proíba o tráfego de aeronaves trafegando por elas. Essas indicações são informadas por meio de cartas de navegação que mostram essas áreas e permitem que o piloto, de maneira manual, possa planejar sua trajetória de voo (FAA, 2016).

Uma outra característica que pode ser observada na Figura 6 são alguns termos relacionados às alturas, em pés, do espaço aéreo, nas diferentes classes do espaço. De uma forma geral, é necessário consultar cartas de navegação para se saber a altura exata e a classe do espaço aéreo em determinada área. Os espaços são divididos em níveis de voo (*flight levels* – FL), como mostrado na Figura 6. Por exemplo, o nível FL600 (60.000 pés) é o limite da troposfera (limite da capacidade de voo das aeronaves de uma forma geral), de forma que os níveis acima dessa altitude não são relevantes para a aviação.

O espaço aéreo classe A começa em 18.000 pés MSL (*Mean Sea Level*), ou 18.000 pés acima da referência do nível do mar, ou a 1.200 pés AGL (*Above Ground Level*) acima do nível do solo, ou seja, acima das montanhas ou do terreno abaixo, independente da altitude do nível do mar. Existe uma diferença entre o AGL, que é

normalmente mais usado nas baixas altitudes, por motivo de se evitar colisões com o solo, e o MSL, que não tem o solo embaixo da aeronave como referência, e sim o nível do mar, o que faz mais sentido em voos em alta altitude e de longa distância, em que a distância com o solo pode mudar, mas a aeronave mantém sempre a mesma altitude.

Nos sistemas de navegação dessas aeronaves, além dos diferentes tipos de espaços aéreos, os pilotos utilizam mapas, chamados de cartas aeronáuticas (*air navigation maps*), que contêm informações sobre o terreno, o espaço aéreo e os pontos de referências (*waypoints*) utilizados para navegação. As aeronaves navegam utilizando esses pontos de referência, seguindo ou atravessando rodovias aéreas conhecidas como aerovias (*airways*).

Pode-se observar, na Figura 7, um exemplo de como uma aeronave se utiliza de torres de rádio para a navegação. Ao posicionar a bússola em 330 graus, e utilizando a torre de rádio (círculo ao centro da imagem) como guia, a aeronave consegue navegar através da linha amarela, que corresponde a sua aerovia desejada.

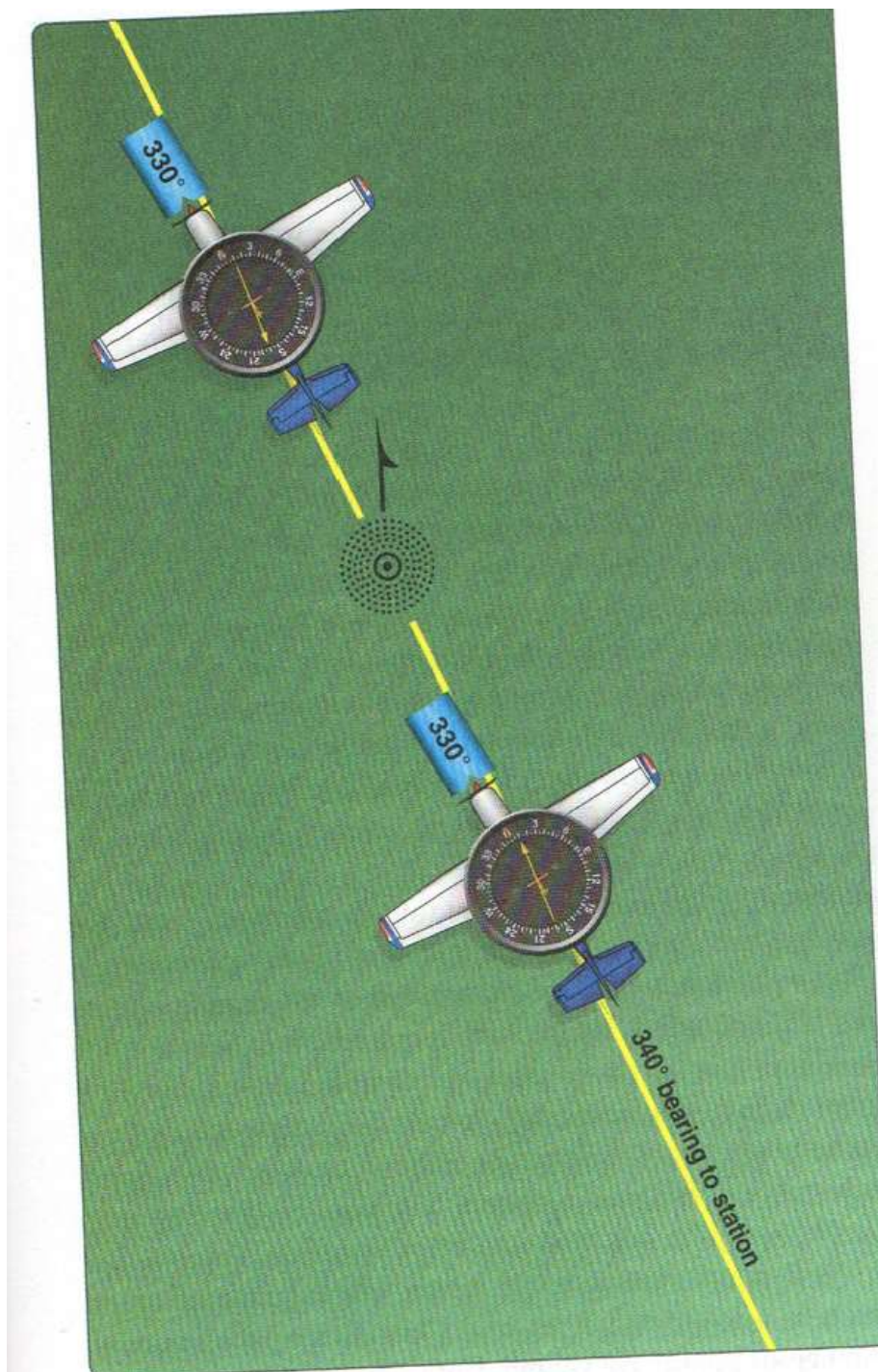


Figura 7 – Aeronave navegando usando uma torre como guia e a sua bússola para seguir o caminho determinado

Fonte: FAA (2016, pág. 16-31)

Pode-se visualizar, na Figura 8, um exemplo completo de uma carta aeronáutica, mostrando todo o espaço aéreo da região Rio-São Paulo.



Figura 8 – Carta aeronáutica Rio-São Paulo

Fonte: DECEA (2022)

Na imagem da Figura 9 encontram-se mais detalhes, dentro desse mesmo espaço aéreo, estando, em azul, as diferentes aerovias, com descrição de nome pontos de referência para a navegação, além de diferentes referências de solo e altitude em cada ponto.

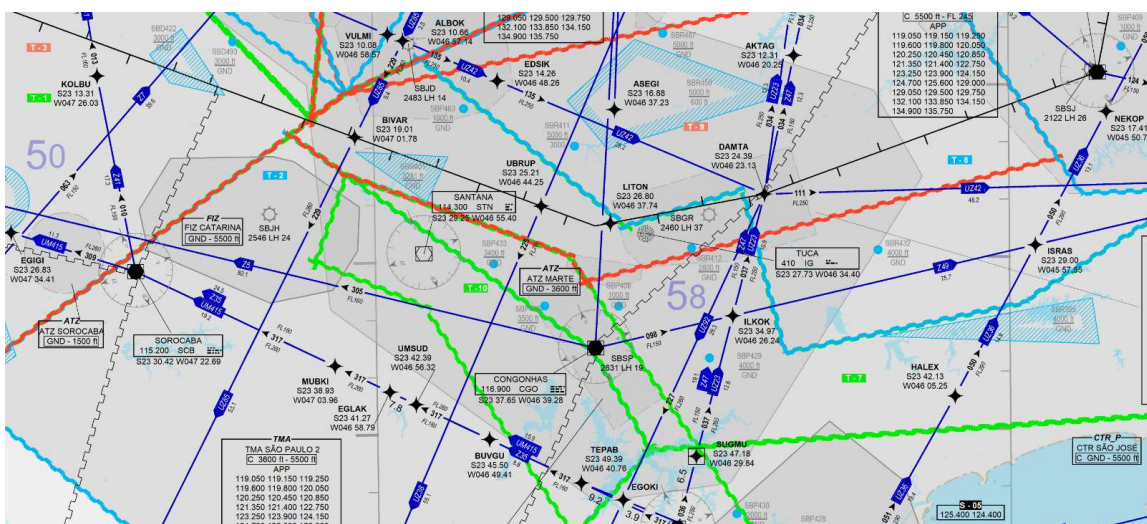


Figura 9 – Detalhe da carta aeronáutica Rio-São Paulo

Fonte: DECEA (2022)

Na sequência, na Figura 10, tem-se uma versão com mais detalhes ainda. Nessa nova imagem pode-se observar, com mais clareza, o Aeroporto de Congonhas, com a orientação de bússola sobre o aeroporto, já que as posições das pistas são indicadas por números referente às direções, em graus, em que cada uma se encontra, além de apresentar indicações de níveis de voo e altura do solo.

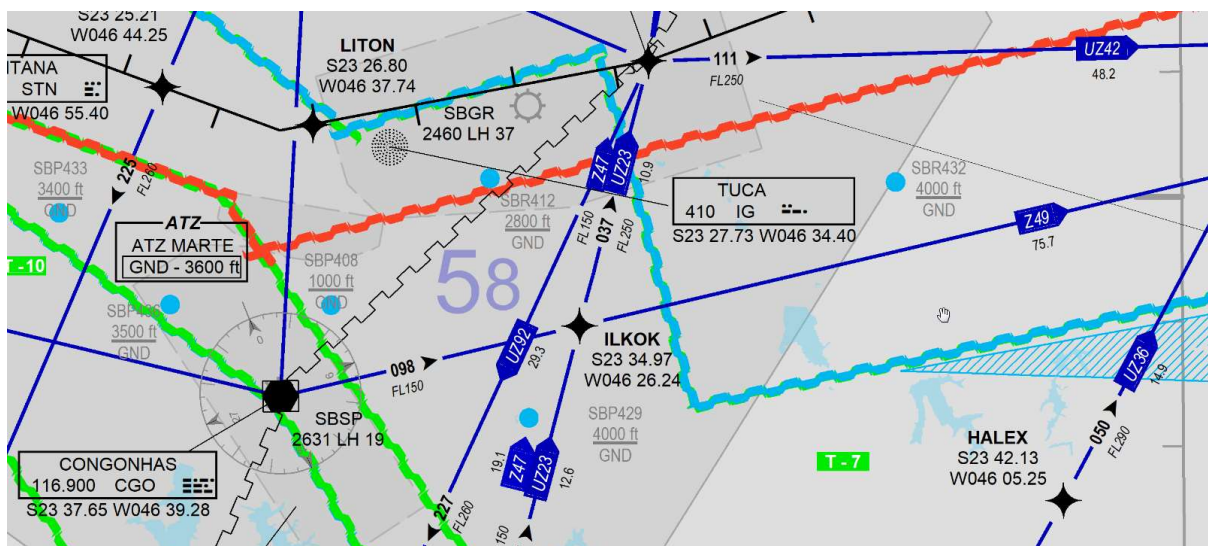


Figura 10 – Detalhe da carta aeronáutica Rio-São Paulo: destaque para o Aeroporto de Congonhas

Fonte: DECEA (2022)

A divisão dos níveis dentro das aerovias também depende da direção para onde está apontada a proa da aeronave com relação à bússola e, dependendo de a quantos graus a proa da aeronave está, é atribuído um nível de voo diferente a essa aeronave, que ela deverá seguir para se evitar conflito com outra aeronave vindo em direção oposta. Isso facilita a divisão do espaço aéreo em setores, o que ajuda a reduzir o número de eventuais conflitos entre aeronaves.

Tem-se, na imagem da Figura 11, a divisão do espaço aéreo em níveis de voo par (EVEN) e ímpar (ODD), segundo a ICAO (*International Civil Aviation Organization*), baseados na direção, em graus, para onde aponta a proa da aeronave. Pode-se notar que nesses dois círculos da Figura 11 existem dois padrões, um para o IFR (regras de voo por instrumentos – *Instrument Flight Rules*) e outro para o VFR (regras de voo visual – *Visual Flight Rules*). Assim, quando se está em voo VFR, e com a proa apontando para um ângulo entre 0° e 179°, atribui-se os níveis ímpares, por exemplo o nível FL15 (referente a altitude de 1.500 pés), e outras aeronaves com

a proa apontando para um ângulo entre 180° e 359° ficariam, por exemplo, com o nível FL25 (referente a altitude de 2.500 pés). Assim, mantém-se uma separação mínima de 1.000 pés entre aeronaves voando em sentidos contrários, que é a separação mínima utilizando-se as regras para o voo do tipo VRF.

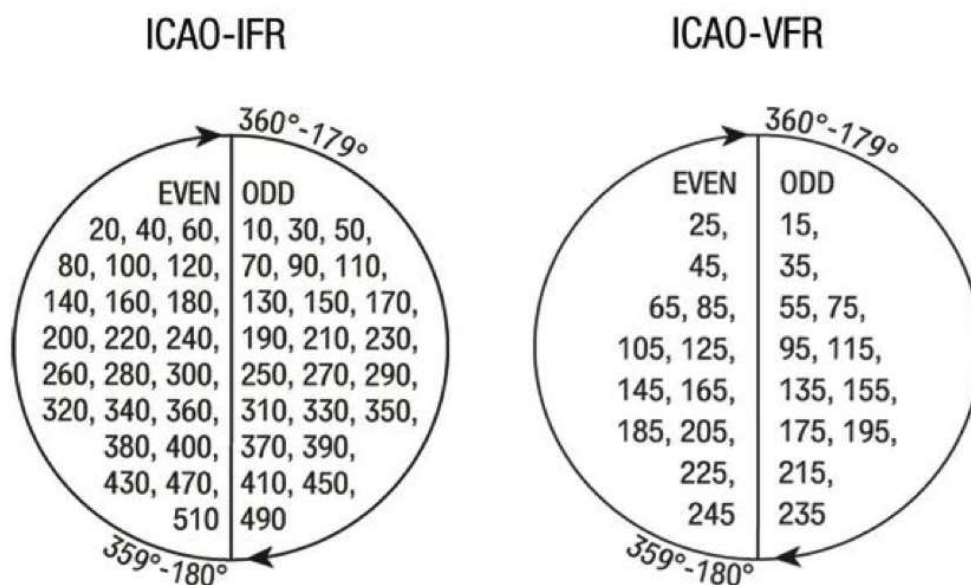


Figura 11 – Divisão de níveis de voo ICAO para voos IFR e VFR

Fonte: FLIGHTCREWGUIDE (2022)

Tem-se, na Figura 12, um exemplo do porquê, nos planos de voo, principalmente com aeronaves de asa fixa, existe a necessidade de o espaçamento estar além da capacidade de manutenção da separação; neste caso, as asas fixas geram vórtices que colocam em turbulências outras aeronaves próximas. Isso evidencia a importância da separação correta e do efeito do vento nas aeronaves.

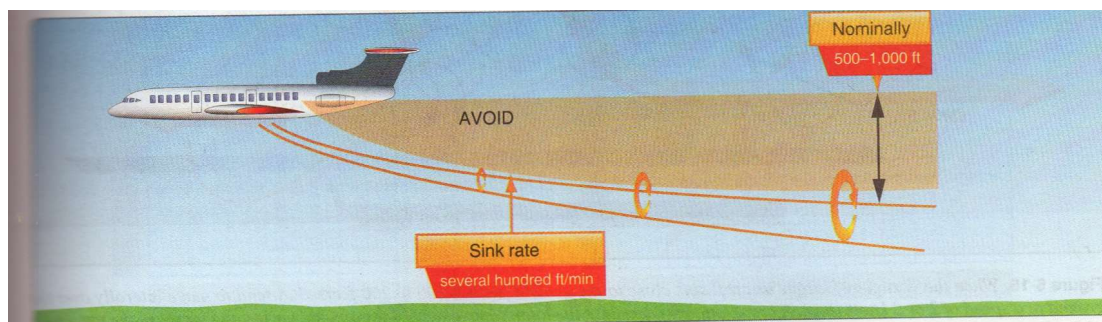


Figura 12 – Exemplo de vórtice formado na asa da aeronave

Fonte: FAA (2016, pág. 5-9)

Finalmente, na Figura 13, tem-se o exemplo de um plano de voo em cima de uma carta aeronáutica. Nesse exemplo, a aeronave sai do ponto A e se desloca até o ponto D, direcionando-se em vários tipos de classes de espaço aéreo e atravessando vários *checkpoints*.

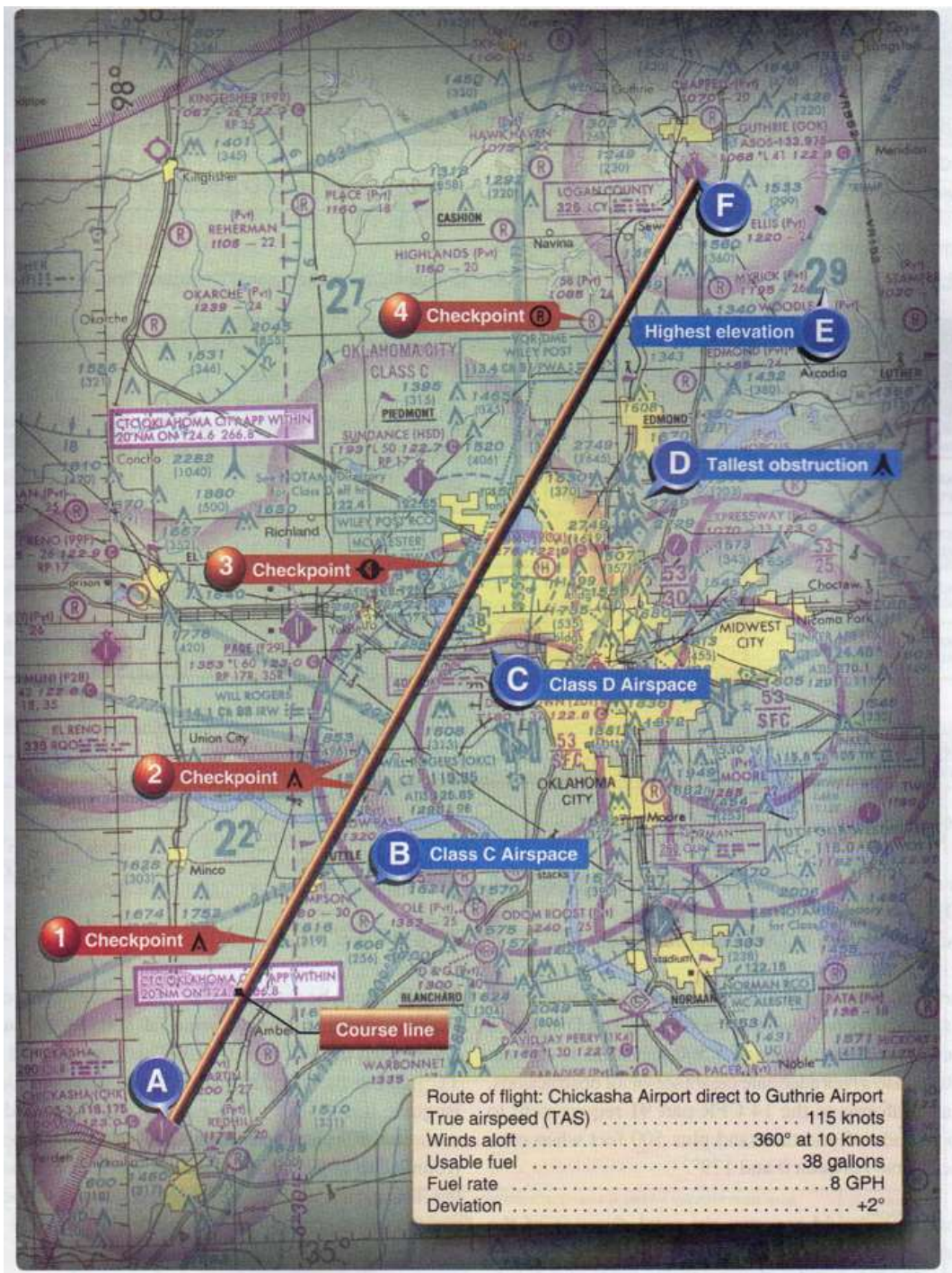


Figura 13 – Exemplo de um plano de voo desenhado sobre uma carta aérea

Fonte: FAA (2016, pág. 16-19)

4.3 UAS *TRAFFIC MANAGEMENT* (UTM) E *URBAN AIR MOBILITY* (UAM)

Pode-se perceber que, no caso da utilização de UAVs totalmente autônomos, existirá uma enorme exigência em termos de complexidade e segurança para a construção dos sistemas que compõem o UAS. Como já visto anteriormente, o UAS corresponde a todos os sistemas que compõem os UAVs e os sistemas do ATM. Com base nisso, o UTM é uma proposta de ATM integrando os sistemas do ATM aos do UAS. Já na proposta do UAM existe um foco maior na mobilidade (integração urbana) e no transporte, apesar de também a iteração dos sistemas ser importante.

Com a grande demanda futura da indústria para a criação de UAVs de uso em áreas urbanas e em outras áreas, existe a preocupação quanto à infraestrutura e tecnologia a serem empregadas. Uma preocupação atual diz respeito a como pode-se fazer o controle do tráfego aéreo dessas aeronaves e avaliar os seus impactos (Lundberg et al., 2018).

Em alguns trabalhos científicos têm-se colocado a necessidade de um sistema de tráfego aéreo para solucionar os problemas encontrados nos UAVs. Tal sistema tem sido chamado de UTM (*UAS Traffic Management* ou *Urban Traffic Management*, dependendo do autor), e trata-se genericamente de um termo para se determinar um sistema de gestão de tráfego incluindo aeronaves autônomas ou RPAS, normalmente visando aeronaves pequenas, ambientes segregados e em baixas altitudes.

Como exemplo, descreve-se em (Glaab et al., 2018) a versão do UTM sendo desenvolvida pela NASA, que incluiu diversos aspectos técnicos de controle de frequências de rádio necessárias para a implantação dessas tecnologias de controle. Um outro aspecto necessário para o UTM, mostrado em (Balachandran, 2017), parte do mesmo projeto da NASA. Existe a necessidade de troca de informações para se gerar trajetórias para as aeronaves, visando garantir a confiabilidade e a segurança do sistema. Na aviação tradicional, existem os sistemas do ADS-B (*Automatic Dependent Surveillance-Broadcast*), que permitem a troca de informações entre aeronaves e outros sistemas, e uma maior automação, independente do piloto, que funciona por uma combinação de troca de informações de trajetórias e decisões automatizadas, o que poderia ser adaptado para o uso em UAVs. Assim, é necessária a divisão do espaço aéreo, seja para a navegação tradicional controlada, ou seja, para a navegação não controlada, criando-se alguns sistemas de controle e regras bem definidas.

Esse sistema permitiria a troca de informações, como aquelas trocadas entre as aeronaves e informações climáticas. Com isso é possível considerar áreas em que é viável a navegação ou considerar áreas com restrições segundo aspectos relatados por boletins meteorológicos, fazendo com que essas informações não sejam úteis somente para a segurança, mas também para a organização otimizada desse sistema.

Têm sido utilizados sistemas de proteção de espaços aéreos, como as “*geofences*” ou “*geo-cercas*”, que são cercas eletrônicas dinâmicas e virtuais, que não existem efetivamente no mundo real, mas que são baseadas em pontos de coordenadas de GPS e determinadas por meio de *software*, de maneira que uma autoridade central possa exercer controle e restringir acessos a determinadas áreas, de maneira a manter o tráfego em um perímetro urbano, de forma organizada e segura (Lundberg et al., 2018).

Sendo assim, normalmente tem-se pensado no uso do UTM como um mecanismo complementar ao ATM tradicional, apesar de muitas tecnologias ainda estarem incipientes para serem colocadas em produção. Pretende-se, então, que esse sistema funcione em todo o espaço aéreo onde não exista ATM (normalmente até 400 pés acima do solo) e em áreas restritas. A proteção do sistema de gerenciamento de tráfego de aeronaves seria controlada com uso de geo-cercas (como, por exemplo, em áreas de pouso e decolagem de aviões) (Lundberg et al., 2018).

A Figura 14 ilustra simplificada um UTM. Apesar disso, na literatura apresentam-se propostas diferentes, principalmente com relação à altura mínima exigida para as aeronaves, notadamente com regras que consideram alturas maiores em regiões com muitos prédios altos. A Figura 14 mostra a proposta da NASA considerando a altura máxima de 400 ft como sendo um ponto ideal de partida para sistema de controle gerenciado dos UAVs. Assim, tanto na proposta elaborada pela FAA (2020), quanto na ilustrada pela Figura 14, pode ser considerado o UAM como qualquer espaço gerenciado acima de 400 ft. Abaixo de 400 ft as aeronaves podem se movimentar com voo livre, exceto em alguns espaços restritos (aeroportos, por exemplo)



Figura 14 – NASA Unmanned Aircraft System (UAS) Traffic Management (UTM)

Fonte: NASA UAM (2021)

Como essa é a proposta que utiliza faixas de altura mais abrangentes e também é a mais aceita por órgãos oficiais, ela será utilizada por este trabalho de pesquisa. No caso desse trabalho a modelagem visando a simulação se dará entre 1.000ft e 3.200ft.

No decorrer do desenvolvimento do UTM houve uma mudança de paradigma a partir do qual começou a ser tratado mais especificamente o UAM (*Urban Air Mobility*), que seria uma evolução de paradigma observado no UTM. Nesse novo paradigma, além do *Air Traffic Control*, busca-se uma visão mais integrada com as necessidades do espaço urbano como solução para o tráfego, considerando a mobilização de encomendas, pessoas, trânsito, pensando não somente nos aspectos de trânsito, como também nos aspectos de pouso, recarga e impacto urbano.

Assim, o problema no desenvolvimento do UAM, além das questões já levantadas, seria a questão do próprio ambiente urbano que cria vários problemas para a navegação de aeronaves. Pode-se citar, neste caso, além das distâncias entre as aeronaves, as distâncias com outros objetos para não colidir, talvez, com construções ou antenas. Além disso, existem outras questões básicas importantes a serem consideradas, como os problemas decorrentes de existirem aeronaves com câmeras e sensores passando perto de janelas de prédios e de casas, criando

problemas com questões de privacidade e de barulho feito pelas aeronaves, além da própria segurança das pessoas e propriedades. Ainda, existem questões a serem consideradas como os túneis de ventos que são criados entre prédios e que poderiam acarretar sérias complicações de segurança de voo dessas aeronaves, principalmente por serem aeronaves muito grandes. Assim, apesar de se ter aparentemente uma grande área de voo teoricamente possível em baixa altitude, o tamanho da área de voo possível real é muito menor do que somente aquela considerando as restrições dos obstáculos visíveis, já que considerando as restrições de segurança e de privacidade já citadas tornam o voo em baixa altitude não possível sem causar riscos ou inconvenientes (Bauranov e Rakas, 2021).

5. MODELO MULTIAGENTE PARA SIMULAÇÃO DE UAVS

Neste capítulo apresenta-se de forma detalhada uma descrição do modelo de simulação desenvolvido nesse trabalho. Este capítulo está dividido em algumas seções para detalhar diferentes áreas do modelo. Inicialmente, existe uma introdução com os principais conceitos utilizados no modelo. Na segunda seção, é feita uma descrição geral do modelo desenvolvido. Na terceira seção, encontra-se uma descrição da interface de controle do modelo desenvolvido, na qual se pode controlar os parâmetros do modelo de simulação e acompanhar os gráficos e os valores das variáveis gerados. Na quarta seção, descreve-se a interface de simulação do Netlogo, no qual é possível visualizar os agentes se locomovendo no ambiente de simulação durante a sua execução. Na quinta seção, encontram-se descrições dos agentes construídos nesse modelo. Na sexta seção, é apresentado o funcionamento dos CBs dentro do modelo desenvolvido e as suas diferentes características. Na sétima seção, descrevem-se os diferentes tipos de trajetória de aeronaves utilizados no modelo desenvolvido. Na oitava seção, descrevem-se os diferentes tipos de parâmetros usados na configuração de entrada do modelo. Na nona seção, são descritos e diferenciados os dois diferentes tipos de separação utilizados no modelo. Na décima seção, descrevem-se os atrasos (*delays*) e porque eles são importantes dentro da simulação. Por fim, na última seção, é descrito o modelo proposto utilizando como referência o protocolo ODD para descrição de modelos multiagentes.

5.1 CONSIDERAÇÕES GERAIS DO MODELO

Neste capítulo, é descrito o modelo desenvolvido neste trabalho de pesquisa. Alguns dos resultados deste trabalho podem ser encontrados em Ferrare et al. (2021b), em que se descrevem algumas características desse modelo, e em Ferrare et al. (2021a), no qual alguns cenários de simulação são escolhidos para a apresentação prática de resultados e de algumas simulações experimentais executadas.

Decidiu-se, neste trabalho, restringir a modelagem do espaço aéreo considerando a região limitada do UAM, e com isso foi considerado que o espaço no qual as simulações são realizadas correspondem à região entre 1.000 ft e 3.200 ft

acima do solo (aproximadamente 305 e 976 metros, respectivamente), isso devido aos fatos citados anteriormente, de conflitos em baixa altitude com relevo ou edificações, e conflitos em alta altitude com outras aeronaves do tráfego aéreo tradicional, criando um espaço de utilização onde se pode ter uma capacidade muito mais bem aproveitada. Essa região simulada é onde as aeronaves executam suas trajetórias. O espaço aéreo acima de 3.200 ft é considerado espaço aéreo regulado, enquanto que o espaço aéreo abaixo de 1.000 ft é considerado espaço aéreo livre. Apesar de o espaço abaixo de 1.000 ft ser considerado “livre”, espera-se que as aeronaves apenas subam e desçam nessa aérea, e não fiquem executando voos de um ponto a outro. Assim, em geral, o espaço aéreo acima de 3.200 ft já possui outros tráfegos e, por outro lado, o espaço aéreo abaixo de 1.000 ft possui obstáculos (prédios, torres). Com a modelagem considerando o espaço acima de 1.000 ft, na prática pode-se ficar acima da maior parte desses obstáculos, como detalhado mais à frente. No modelo de simulação proposto neste trabalho, as aeronaves irão entrar e sair do espaço de simulação sempre pela parte de baixo do espaço simulado, oriundas de algum “heliporto”; desta forma, assume-se que essas aeronaves já estão voando no momento que entram no espaço da simulação. Ainda, as aeronaves não podem atravessar as partes laterais e superior do espaço aéreo simulado.

As simulações são, então, executadas múltiplas vezes. Como as execuções geram trajetórias de maneira estocásticas, a execução do modelo está condicionada a uma variação a cada execução. Dessa forma, pode-se analisar não o resultado individual de uma simulação única, mas sim o padrão da distribuição do conjunto de resultados de diversas simulações, o que permite se ter uma melhor noção do impacto da variação dos parâmetros de entrada nos resultados obtidos (Thiele et al., 2014).

Para as simulações realizadas neste trabalho, utilizou-se um *notebook* padrão *quad-core*, com 16GB de memória RAM, para execução das simulações do modelo, utilizando-se diferentes parâmetros. O *software* Netlogo versão 6.x., destinado à simulação de modelos multiagentes, foi o escolhido para as simulações realizadas neste trabalho, pois a linguagem Netlogo oferece um ambiente com uma curva de aprendizado muito boa, além de ser poderosa e versátil, sendo capaz de simular ambientes físicos ou modelos de ciências sociais.

No ambiente acima de 3.200 ft do solo, existiriam problemas e restrições para o voo de aeronaves autônomas, pois acima dessa altitude o espaço aéreo é regido por regras bem definidas e, a partir desse limite, trafega por ele a aviação tradicional

(aviões, helicópteros). E para o caso de os veículos autônomos trafegarem nesse mesmo espaço, seriam necessárias regras especiais para esse tipo de ambiente (no caso de se trabalhar com ambientes não segregados). Por outro lado, o ambiente abaixo de 1.000 ft tem outros desafios que necessitariam de regras diferentes, principalmente pelo fato de haver o risco de colisão dos veículos autônomos com obstáculos naturais ou artificiais (prédios, antenas, etc.), e pelo fato de esses obstáculos criarem "túneis de vento", tornando a navegação de UAVs mais problemática e perigosa, susceptível a acidentes.

Apesar de se considerar o UAM como qualquer espaço acima de 400 ft, neste trabalho e na abordagem desta pesquisa decidiu-se modelar e construir um modelo abrangendo apenas uma fração desse espaço (espaço acima de 1000 ft), que foi o espaço considerado mais relevante para as simulações, como visto anteriormente no início do capítulo.

Por isso, no modelo construído neste trabalho, os *drones* apenas navegam entre 1.000 ft e 3.200 ft. Abaixo de 1.000 ft, o espaço aéreo só seria usado para a decolagem e para o pouso final de cada *drone*. Essa condição de contorno não se mostra relevante para as simulações realizadas, além de poder acarretar problemas de navegação dos *drones* navegando entre prédios, devido a correntes de ar e obstáculos, enquanto que na área acima de 3.200ft existem potenciais conflitos com outros tráfegos. Tudo isso, ainda, poderia gerar um maior consumo de energia para viagens não tão longas dentro de um ambiente urbano.

Na imagem da Figura 15, pode-se observar 3 divisões do espaço aéreo: a) a primeira, uma região do espaço aéreo não controlado, abaixo de 1.000 ft; b) a segunda, corresponde à região de validade do UAM (corresponde à região dos limites considerados no modelo deste trabalho), entre 1.000 ft e 3.200 ft; e c) a terceira, que diz respeito à região do espaço aéreo sob o controle do ATM, acima de 3.200 ft.

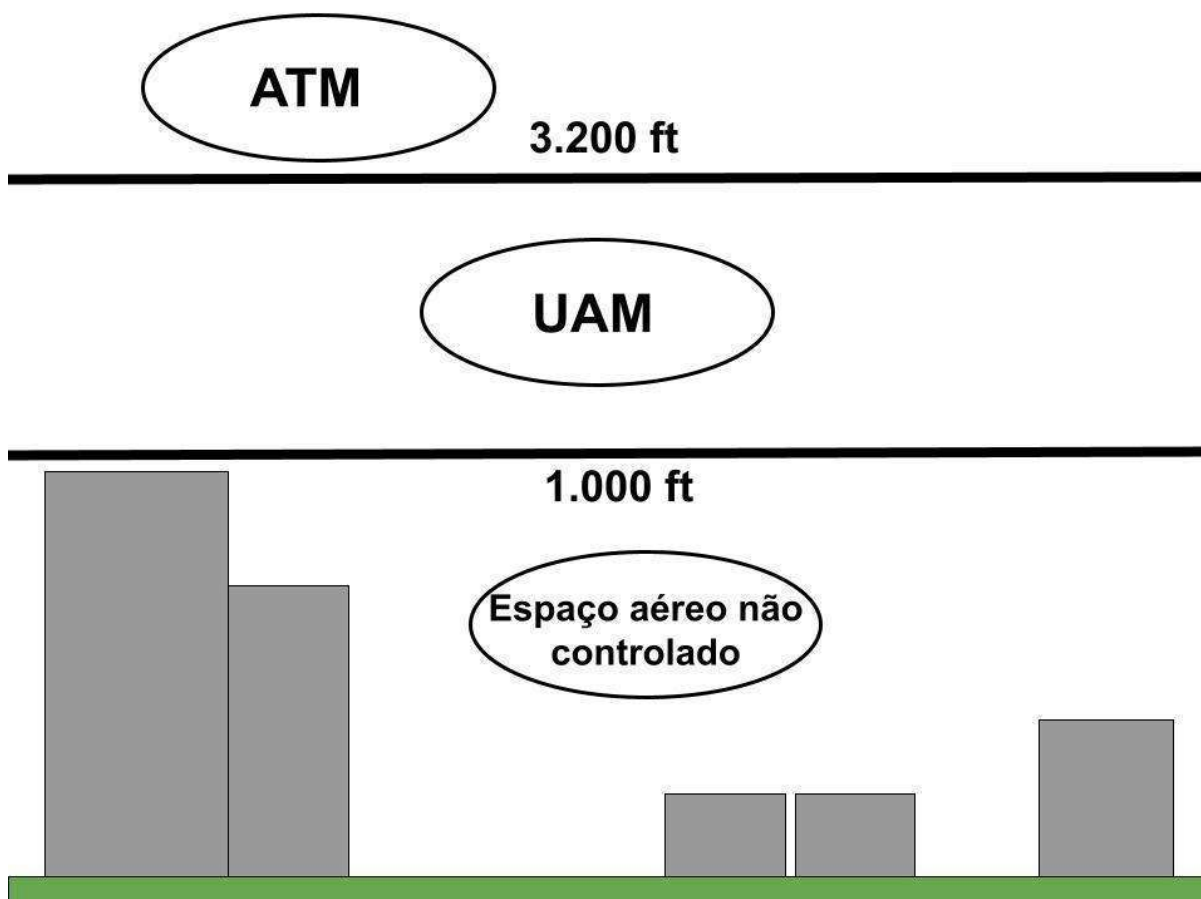


Figura 15 – Divisão do espaço aéreo considerada neste trabalho, com destaque à região do UAM

Fonte: Autor (2022)

Pode-se destacar como uma contribuição desse trabalho a utilização de ideias do controle do tráfego aéreo tradicional no UAM, algo que normalmente não é explorado na maioria dos trabalhos de pesquisa consultados. A própria regulamentação preliminar da FAA (2020) tende ao uso de um sistema que regulamenta apenas “corredores” de VTOLs em níveis mais altos, e não regulamenta regras específicas quando as aeronaves trafegam em baixas altitudes, deixando parte significativa da operação dessas aeronaves ser determinada para casos específicos ou a cargo dos operadores das aeronaves, apesar de muitas das ideias da aviação tradicional já estarem testadas e validadas há muitos anos.

No modelo de UAM proposto neste trabalho foram criadas subdivisões do espaço para permitir o tráfego organizado dos UAVs, minimizando os riscos de colisão entre eles, conforme ilustrado na Figura 16. Nessa figura, as subdivisões criadas dentro do modelo de UAM proposto possuem espaçamento de 200 ft entre as

subdivisões adjacentes, e essas divisões correspondem às faixas de navegação dos UAVs.

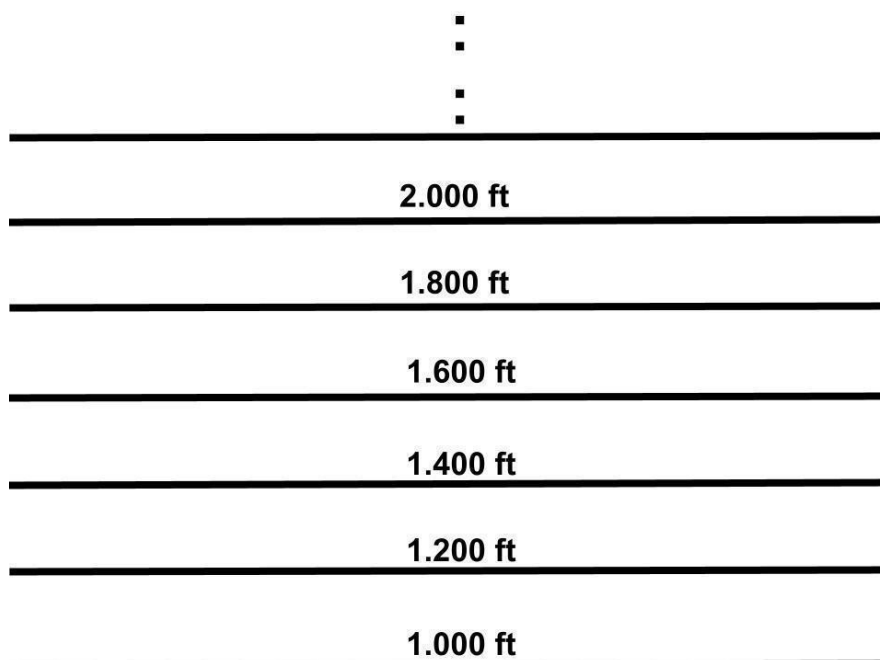


Figura 16 – Subdivisão de altitude do UAM considerada

Fonte: Autor (2022)

Propõe-se neste trabalho, como destacado na Figura 17, uma divisão organizada para o tráfego de UAVs no UAM, considerando a direção da aeronave. Assim como ocorre no espaço aéreo da aviação tradicional, foi feita uma divisão do espaço do UAM em que aeronaves, partindo de certo ponto, possam ocupar apenas algumas faixas do espaço aéreo, de acordo com a sua direção. Essa divisão é feita de tal forma que as faixas que admitem aeronaves vindas de certa direção estão alternadas com faixas que admitem aeronaves trafegando na direção oposta. Assim, cada uma das faixas de tráfego de aeronaves admite aeronaves trafegando em direção contrária às aeronaves presentes em faixas vizinhas (acima ou abaixo). Como o propósito deste trabalho tem o objetivo de ter tanto aeronaves pequenas para transporte de pacotes, quanto aeronaves médias para o transporte de pessoas e cargas maiores, foi considerado esse espaçamento mais estreito do que na aviação tradicional, mas que não é muito restritivo a ponto de impossibilitar o transporte de cargas maiores. Essa é a mesma ideia utilizada nas aerovias do tráfego aéreo tradicional, visando à redução de conflitos entre aeronaves, o que apresenta como vantagem a redução da probabilidade de aeronaves trafegando em rota de colisão frontal.

Conforme a Figura 18, essa divisão do espaço aéreo depende do ângulo da proa das aeronaves (parte frontal) em relação ao ponto de referência, sendo então criadas duas faixas de possíveis altitudes para cada orientação da proa (entre 0° e 180° e entre 180° e 360°). Antes de iniciar sua navegação, a aeronave “escolhe” aleatoriamente (para garantir uma distribuição justa) uma altitude, a qual ela tentará usar por todo o percurso.

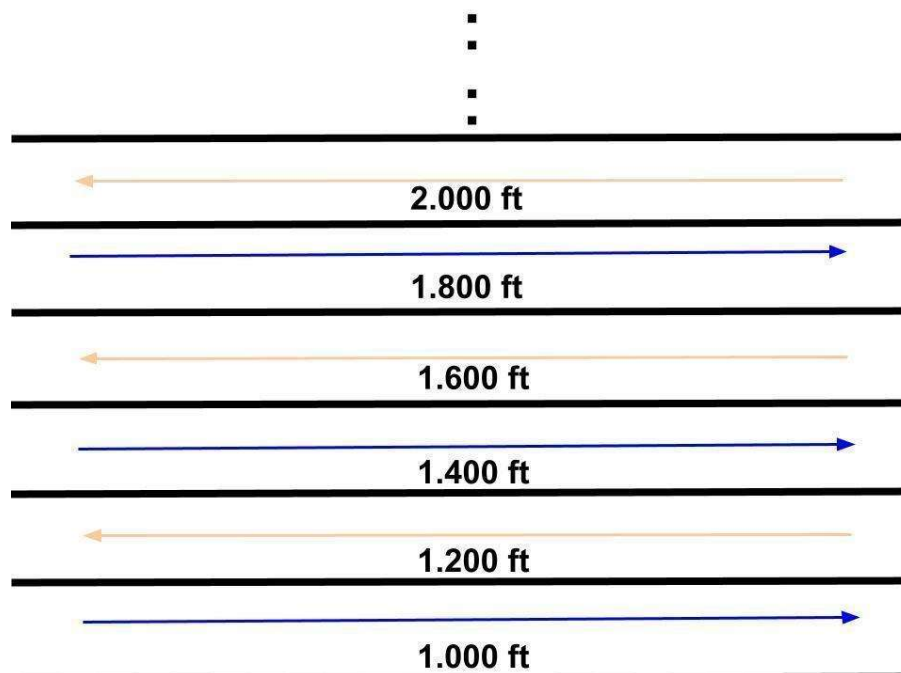


Figura 17 – Faixas de diferentes altitudes das aeronaves em função de sua altitude

Fonte: Autor (2022)

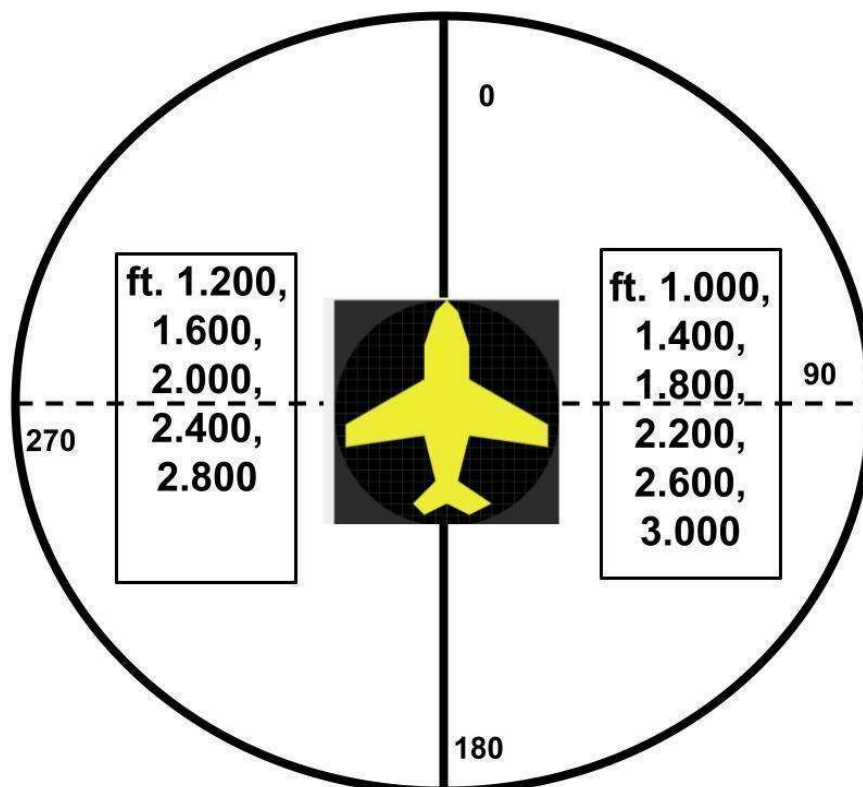


Figura 18 – Faixas de diferentes altitudes das aeronaves em função de sua direção

Fonte: Autor (2022)

Como é possível observar na imagem da Figura 19, ao se usar medidas com referência ao solo, como uma linha 1.000 ft acima do solo, esta pode variar em altitude de acordo com a topografia do local. Para fins do modelo elaborado neste trabalho, consideram-se as áreas envolvidas completamente planas, e como são consideradas divisões quadradas de diferentes tamanhos da área envolvida, considera-se ser possível construir subdivisões em áreas planas, desconsiderando-se eventuais variações menores de altitude.



Figura 19 – Linha de 1000 ft acima do solo

Fonte: Autor (2022)

No ambiente de simulação também é possível controlar o tamanho do campo de simulação e, a princípio, foram modelados dois espaços de dimensões: um de 15 NM x 15 NM e outro de 30 NM x 30 NM (NM – milha náutica). Deve-se observar que 1NM corresponde a 1,852 km e, portanto, têm-se as respectivas áreas de 27,78 km x 27,78 km (aproximadamente 771 km²) e 55,56 km x 55,56 km (aproximadamente 3.087 km²), que correspondem a áreas equivalentes a uma grande cidade ou um grande centro urbano. A título de exemplo, a cidade de São Paulo tem uma área de 1.521,202 km² (IBGE, 2022), que não é perfeitamente um cubo, mas sim um paralelepípedo. Desta forma, as áreas consideradas nas simulações realizadas podem cobrir uma área metropolitana dessa magnitude. Na Figura 20, tem-se um mapa em escala da cidade de São Paulo e da sua região metropolitana (macrorregião 6, cercada por outras macrorregiões), enquanto que na Figura 21 tem-se o mesmo mapa, em escala, com dois quadrados vermelhos sobrepostos, mostrando como as áreas quadradas de 15 NM e 30 NM se encaixariam em uma cidade ou um centro urbano como São Paulo.

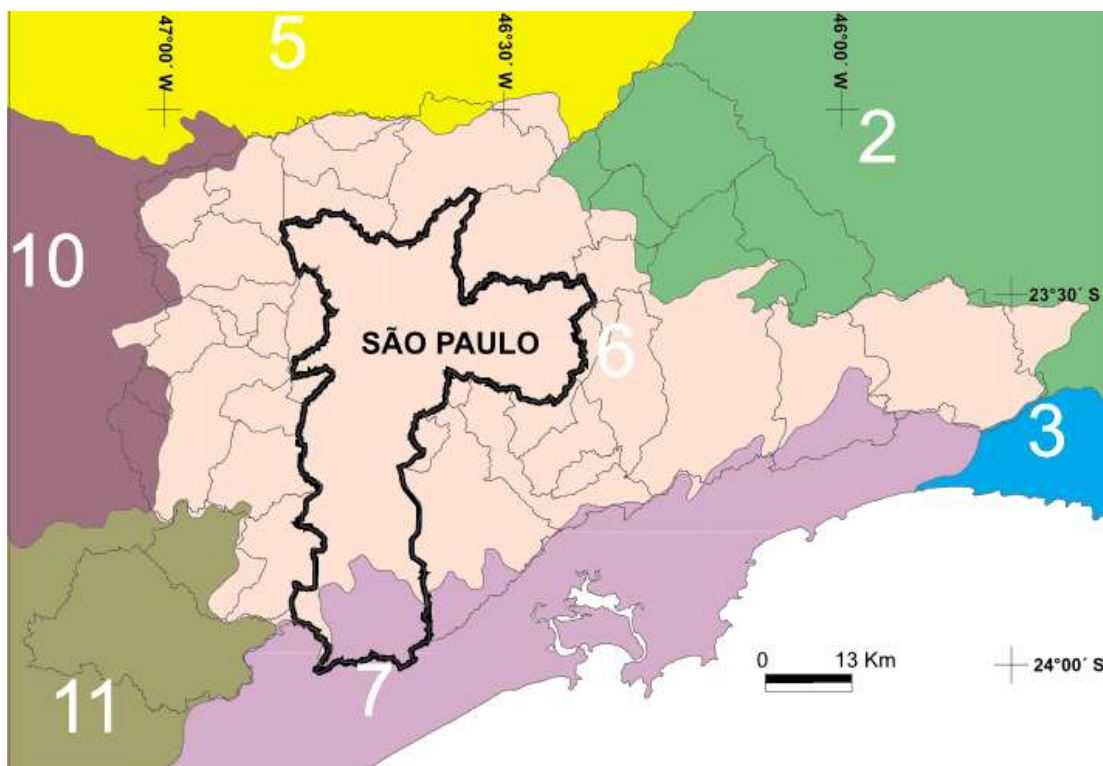


Figura 20 – Mapa da cidade de São Paulo e macrorregiões correlatas, em escala

Fonte: Prefeitura de SP (2022)

Todas as viagens executadas por um eVTOL nas simulações são realizadas dentro de um desses dois quadrados, denominados de ambiente de simulação. Cada viagem deve ser sempre iniciada e terminada dentro desse espaço. Portanto, as aeronaves simuladas não podem entrar ou sair do espaço lateralmente, mas somente por baixo da região simulada, quando estão pousando ou decolando de um local de pouso.

O ambiente de simulação (de 15 NM x 15 NM ou 30 NM x 30 NM, com altura variando entre 1.000 ft e 3.200 ft), por sua vez, é subdividido em quadrículas tridimensionais de base de 0,5 NM x 0,5 NM, por 200 ft de altura, tendo um formato de paralelepípedo, mas usando o nome de quadrícula para simplificar visualmente. Assim, o ambiente tem 30x30 ou 60x60 quadrículas no plano, e 11 quadrículas de altura, totalizando 9.900 ou 39.600 quadrículas, conforme os dois casos citados.

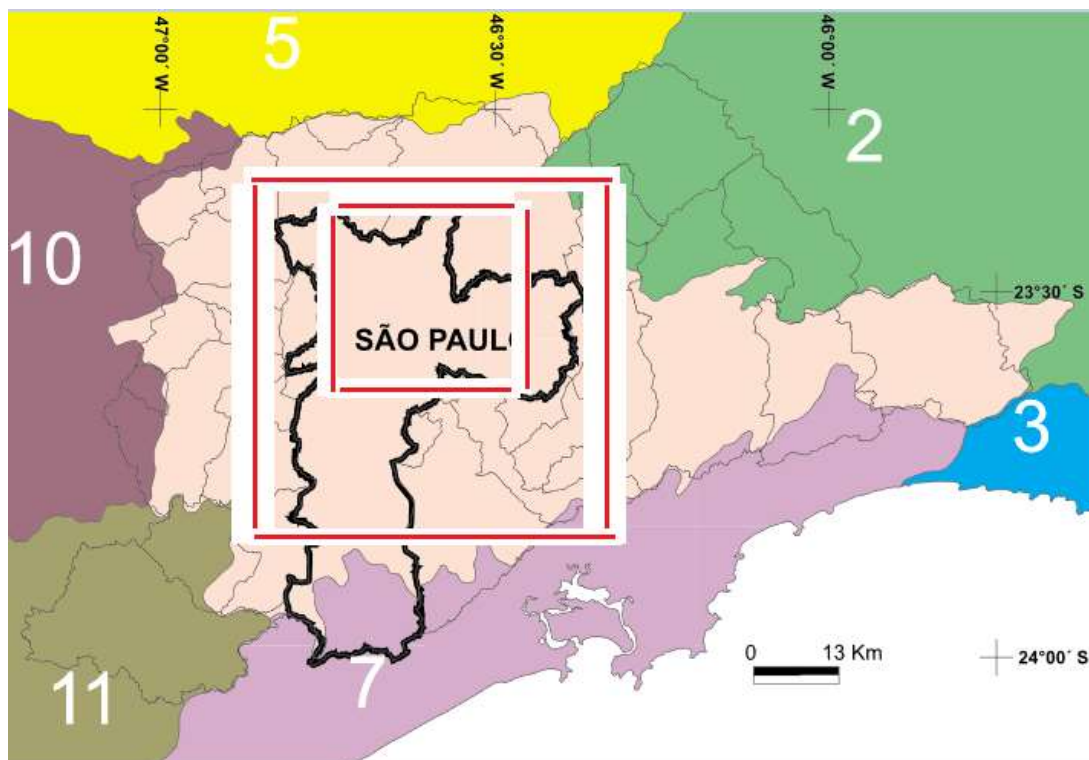


Figura 21 – Áreas quadradas de simulação sobre o mapa da cidade de São Paulo e sobre a macrorregião da cidade São Paulo

Fonte: Autor (2022), Adaptado de Prefeitura de SP (2022)

A quadrícula é a menor subdivisão de espaço no modelo, onde se considera que duas aeronaves em uma mesma quadrícula correspondem a um cenário de uma possível colisão, já que o veículo pode estar em qualquer ponto desse espaço devido a instabilidades das manobras do veículo para a correção de erros que podem acontecer, por exemplo, com pequenas movimentações devido à força dos ventos, ou quando a aeronave está subindo ou descendo, quando pode descrever uma espiral dentro dessa quadrícula para otimizar energia. A Figura 22 apresenta a representação de uma quadrícula, conforme adotada neste modelo.

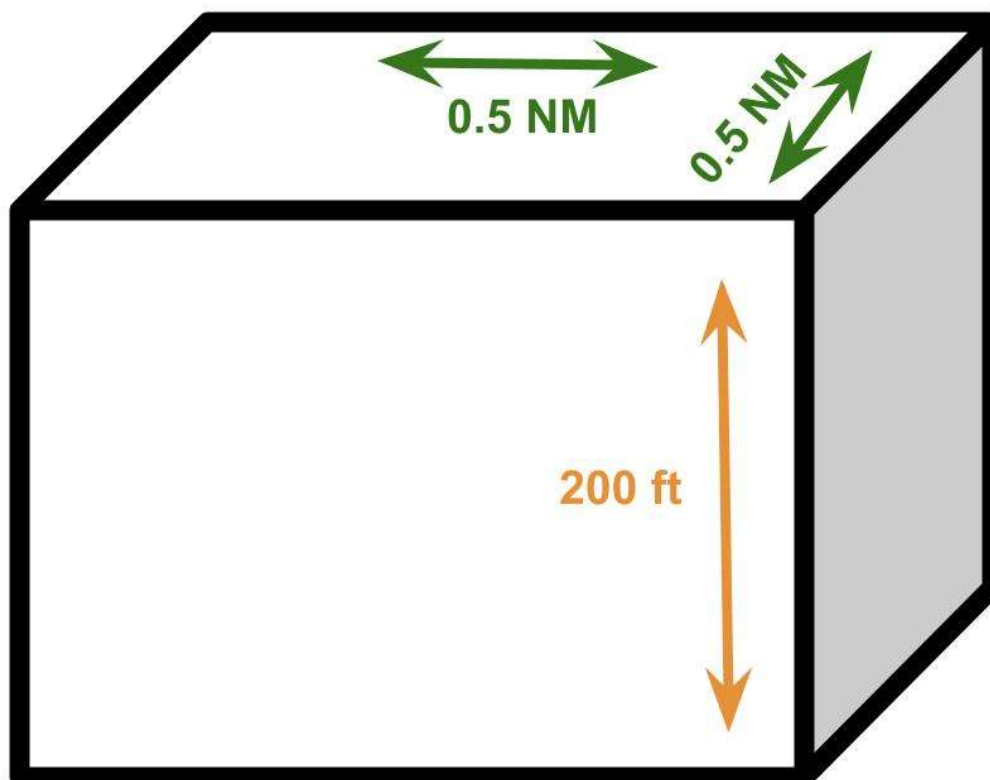


Figura 22 – Representação esquemática de uma quadricula

Fonte: Autor (2022)

Foi considerado, durante o desenvolvimento do modelo proposto nesta pesquisa, que o “combustível” (no caso, a eletricidade alimentando as baterias do eVTOL) da aeronave seria suficiente para o trajeto por ela realizado, tendo a aeronave autonomia suficiente para cumprir o percurso para ela traçado. Além disso, pode-se, nas simulações realizadas, acompanhar o desempenho das aeronaves, verificando-se a distância percorrida e o tempo que a aeronave levou para cumprir a sua trajetória. Com isso, é possível se estimar a quantidade de energia restante (que pode ser considerado “combustível” ou carga de bateria) e a autonomia que essas aeronaves deveriam ter para cumprir, com segurança, essas trajetórias.

5.2 DESCRIÇÃO DO COMPORTAMENTO DO MODELO

Na simulação, as aeronaves vão subir verticalmente até a altitude alvo, então navegar nessas altitudes até o destino e, por fim, descer na vertical. Durante esse

processo, a aeronave coloca um alerta de bloqueio para o espaço que ela vai usar; com isso as aeronaves nos níveis abaixo vão desviar dessa coluna de espaço. De acordo com a descida da aeronave, ela libera as quadrículas pelas quais ela já passou.

A viagem típica (ou trajeto) de uma aeronave passa por diversas fases. Esse trajeto tem 8 etapas, conforme ilustrado pelas figuras de 23 a 30, e retrata a situação padrão em que uma aeronave parte do seu ponto inicial pousada, decola verticalmente (podendo realizar uma espiral), desloca-se horizontalmente e um altitude definida e, por fim, pousa verticalmente (podendo também realizar um trajeto em espiral). Assume-se, nessa sequência, que a aeronave não precisa fazer desvios ou mudanças na rota em relação a sua intenção inicial (os desvios e os conflitos são explicados na próxima seção, e complementam esta explicação base).

- **Etapa 1:** A aeronave (UAV) está pousada nas coordenadas do seu ponto de partida, conhece as coordenadas do seu ponto de destino (marcada com um X), e decide em qual nível de altitude fará seu trajeto rumo ao destino (Figura 23);

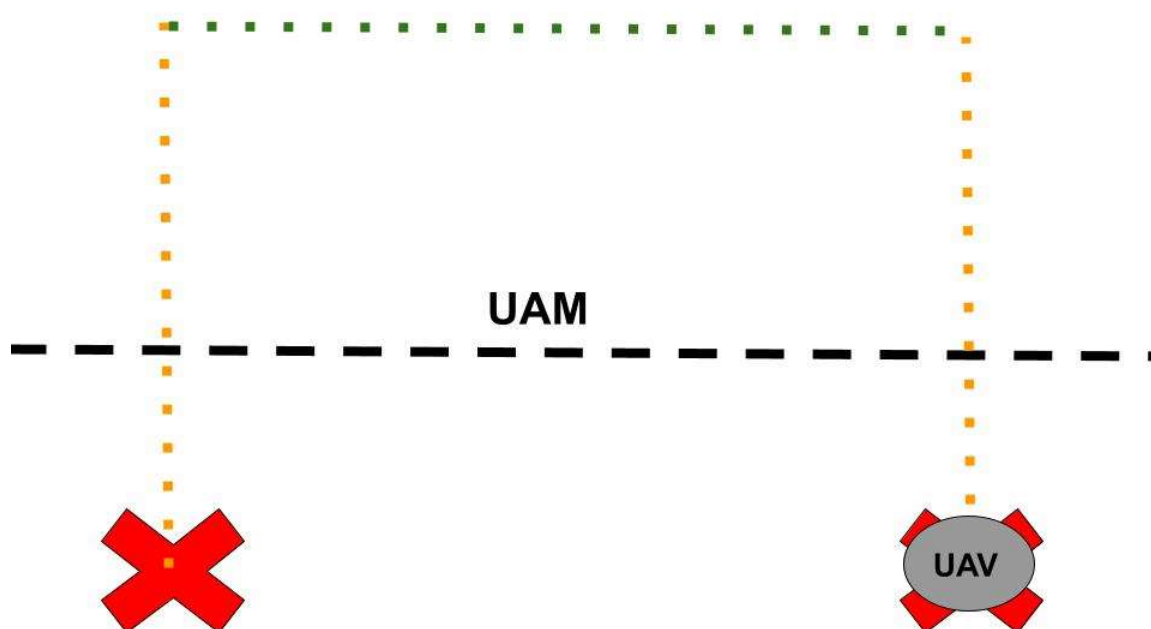


Figura 23 – Etapa 1: Aeronave antes da partida

Fonte: Autor (2022).

- **Etapa 2:** A aeronave (UAV) solicita o bloqueio de todas as quadrículas na vertical pelas quais irá passar durante o processo de subida rumo ao nível de altitude desejado (Figura 24);

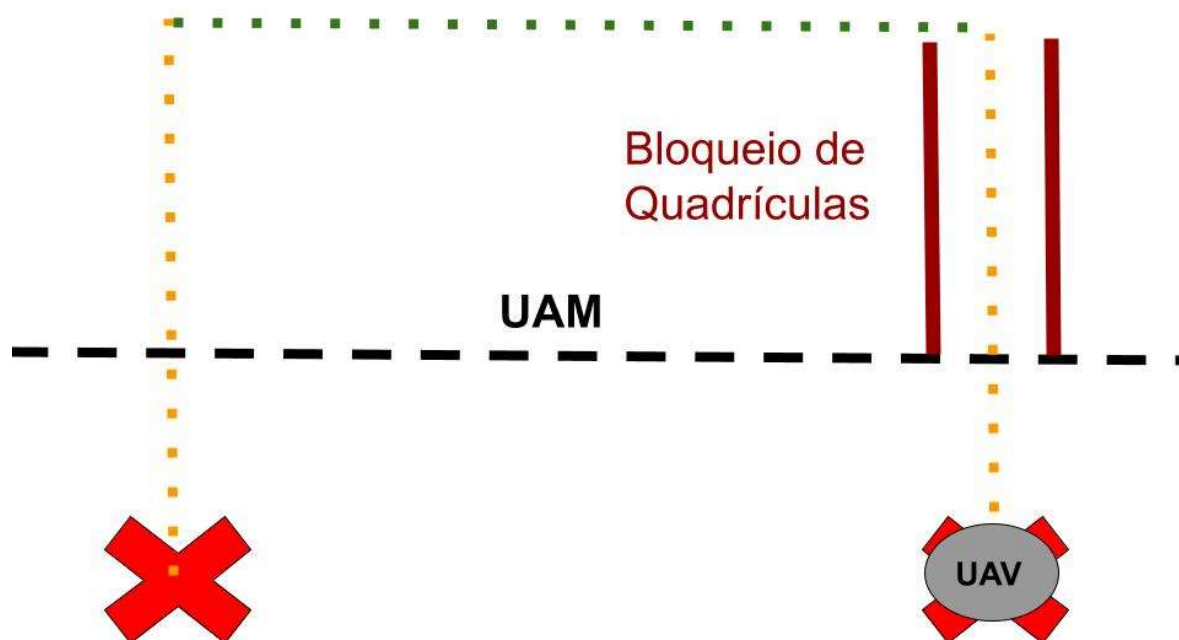


Figura 24 – Etapa 2: Bloqueio de quadrículas antes da subida

Fonte: Autor (2022).

- **Etapa 3:** A aeronave (UAV), enquanto efetua a subida do local de pouso até o nível de altitude desejado, desbloqueia as quadrículas pelas quais já passou (Figura 25);

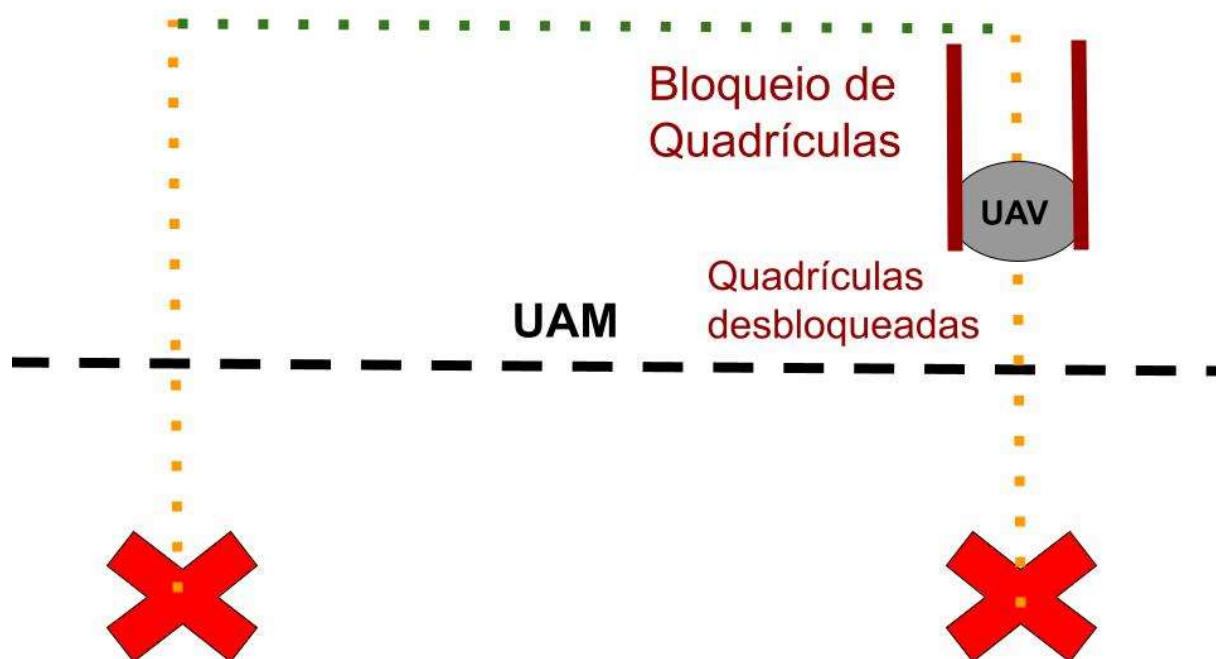


Figura 25 – Etapa 3: Desbloqueio de quadrículas durante a subida

Fonte: Autor (2022)

- **Etapa 4:** A aeronave (UAV), após atingir o nível de altitude desejado, desbloqueou, sucessivamente, todas as quadrículas abaixo dela (Figura 26);

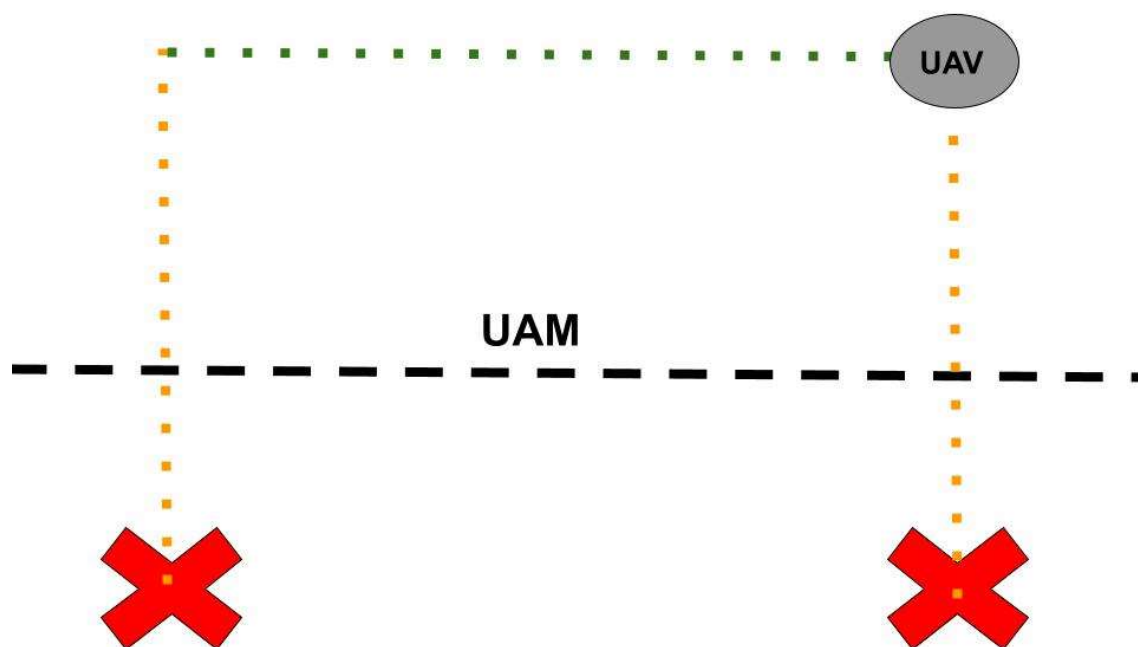


Figura 26 – Etapa 4: Aeronave atinge altitude alvo

Fonte: Autor (2022)

- **Etapa 5:** A aeronave (UAV), já tendo atingido o nível de altitude desejado, navega horizontalmente até o ponto de destino (Figura 27);

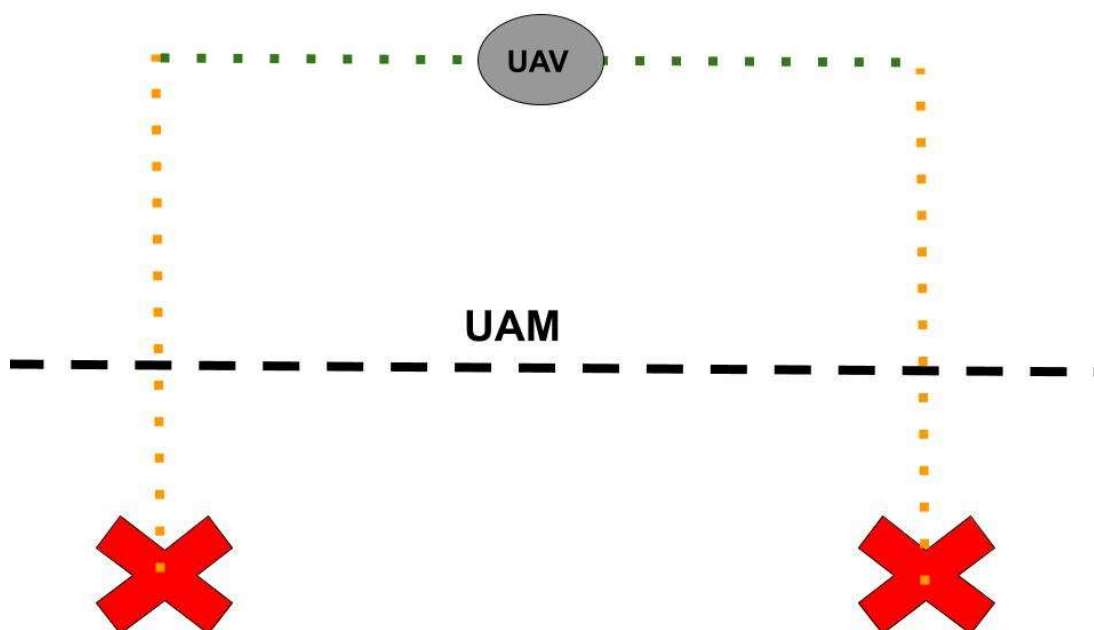


Figura 27 – Etapa 5: Navegação horizontal normal da aeronave

Fonte: Autor (2022)

- **Etapa 6:** A aeronave (UAV), após navegar na horizontal, chega em cima de seu destino e faz o bloqueio das quadrículas abaixo dela (Figura 28);

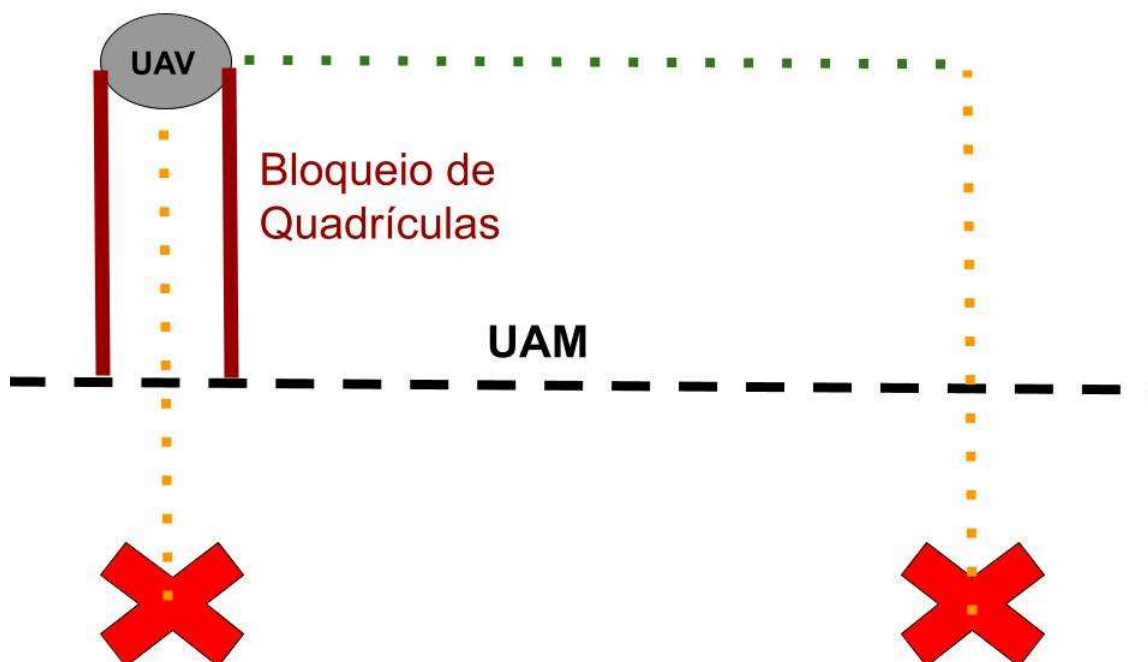


Figura 28 – Etapa 6: Bloqueio de quadrículas para a descida

Fonte: Autor (2022)

- **Etapa 7:** Enquanto a aeronave (UAV) efetua a descida, faz o desbloqueio das quadrículas acima dela pelas quais já passou (Figura 29);

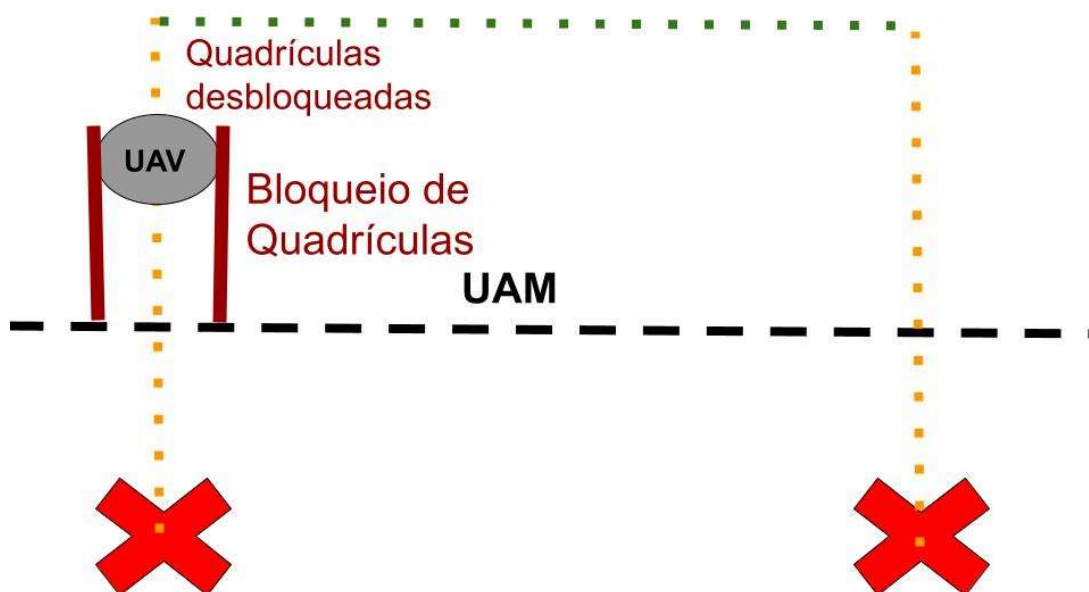


Figura 29 – Etapa 7: Desbloqueio de quadrículas durante a descida

Fonte: Autor (2022)

- **Etapa 8:** A aeronave (UAV) pousa no seu destino, concluindo sua missão (Figura 30).

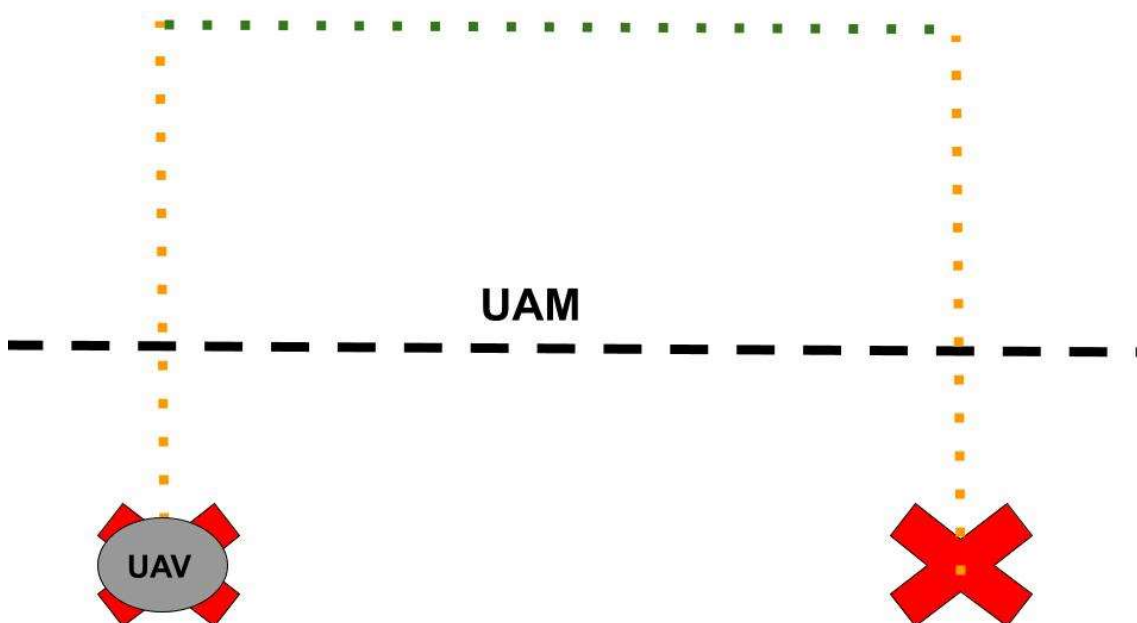


Figura 30 – Etapa 8: Aeronave pousa no seu destino

Fonte: Autor (2022)

O modelo desenvolvido neste trabalho considera três tipos de obstáculos que uma aeronave pode encontrar durante a sua trajetória:

- 1- Formações climáticas de nuvens, como Cumulonimbus (CBs);
- 2- Outras aeronaves;
- 3- Células bloqueadas: quadrículas bloqueadas pela subida ou descida de outras aeronaves.

Durante sua trajetória horizontal, a aeronave pode ter que realizar um dos três tipos de desvio de obstáculos implementados no modelo, devendo ser destacado que os desvios ocorrem independentemente do tipo de obstáculo. O sistema verifica qual tipo de ação é viável e tenta executá-la, conforme a ordem de prioridade descrita na sequência a seguir. Esse modelo não considera que a aeronave vai ter algum tipo de pane durante o trajeto, assumindo que a aeronave sempre vai concluir sua missão.

1- Desvio pelo lado direito: Conforme ilustrado na Figura 31, a aeronave desvia do obstáculo pela direita, sendo que em (1) a aeronave percebe o obstáculo à sua frente; em seguida, em (2), a aeronave virá para a direita e, assim que ela entra em uma quadrícula para o lado direito já sem a interferência do obstáculo, ela restaura a sua direção anterior (3); posteriormente, ela segue até um ponto onde não existe mais a influência do obstáculo que agora encontra-se à sua esquerda (4); então ela faz a mesma manobra para retornar (5); e, por fim, volta à sua direção original (6). Deve-se destacar que não existe a opção de desvio pelo lado esquerdo, por questão de padronização, para evitar que, por exemplo, duas aeronaves em rota de colisão sempre busquem uma alternativa que não gere um novo conflito, como ocorre no tráfego aéreo tradicional.

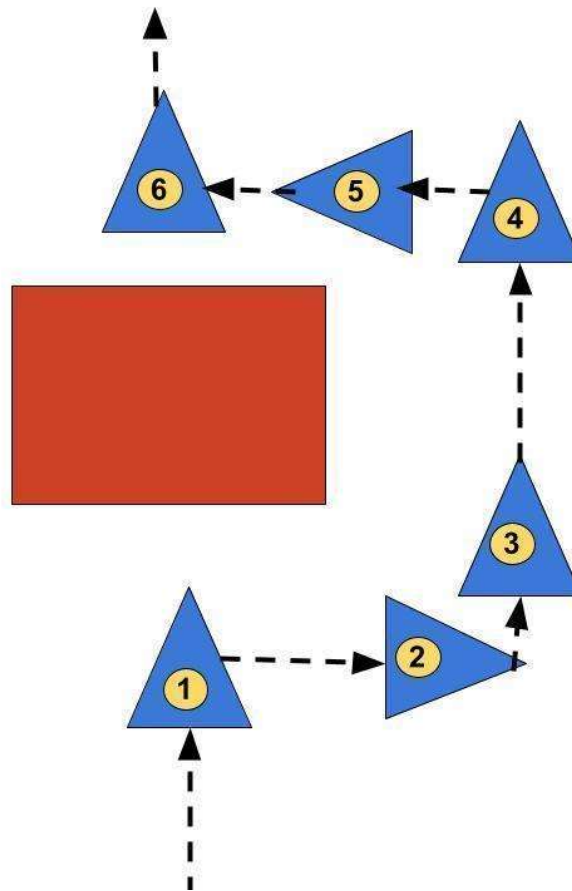


Figura 31 – Desvio de obstáculo pelo lado direito

Fonte: Autor (2022)

2- Desvio por baixo: Conforme a Figura 32, a aeronave percebe a presença do obstáculo (1); desce até um nível abaixo que não possua mais a influência do obstáculo, respeitando a segregação de níveis pela direção da proa, conforme Figura 18 (2); segue na mesma direção original (3); chega em um ponto em que ultrapassou o obstáculo (4); e, por fim, sobe novamente para nível de altitude original (5), mantendo a também direção original. Deve-se destacar que não existe a opção de desvio por cima, por questão de padronização.

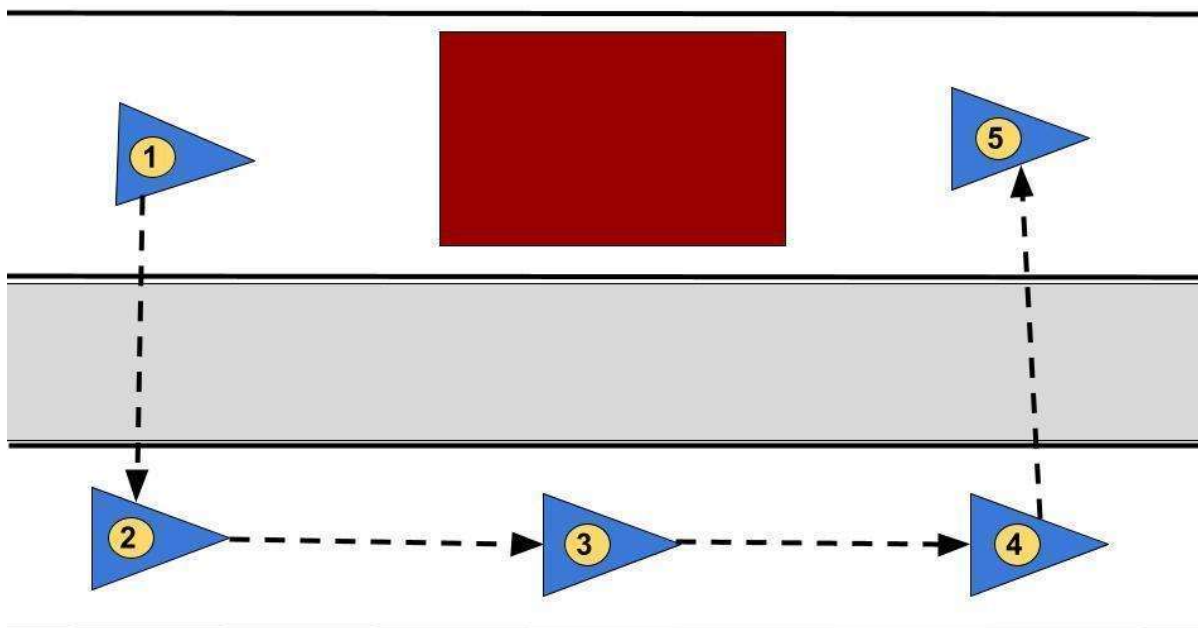


Figura 32 – Desvio de obstáculo por baixo

Fonte: Autor (2022)

3- A aeronave permanece parada: A aeronave permanece em voo pairado ou realizando pequenos deslocamentos em semicírculos, dentro da mesma quadrícula, aguardando as outras aeronaves se movimentarem, visando abrir uma janela de oportunidade para ela poder seguir viagem.

5.3 INTERFACE DE CONTROLE DO NETLOGO

No capítulo 3, foram apresentadas uma introdução aos sistemas baseados em agentes, aos sistemas multiagentes e à linguagem Netlogo. Nesta seção detalha-se o uso dessa ferramenta para modelar o sistema desenvolvido em alto nível. O código-fonte completo encontra-se presente no Apêndice B deste trabalho.

A tela principal do modelo no Netlogo é mostrada na Figura 33, onde se pode observar o “painel de controle” criado para o sistema concebido. Nesse painel, estão presentes as entradas de parâmetros e as saídas produzidas pelo modelo, além de gráficos e botões que permitem realizar ações sobre o sistema, como iniciar, pausar, etc.. É possível observar, também, nas imagens da Figura 34 e da Figura 35, detalhes de como esses valores são apresentados na interface de saída, permitindo o acompanhamento da dinâmica da execução do modelo.

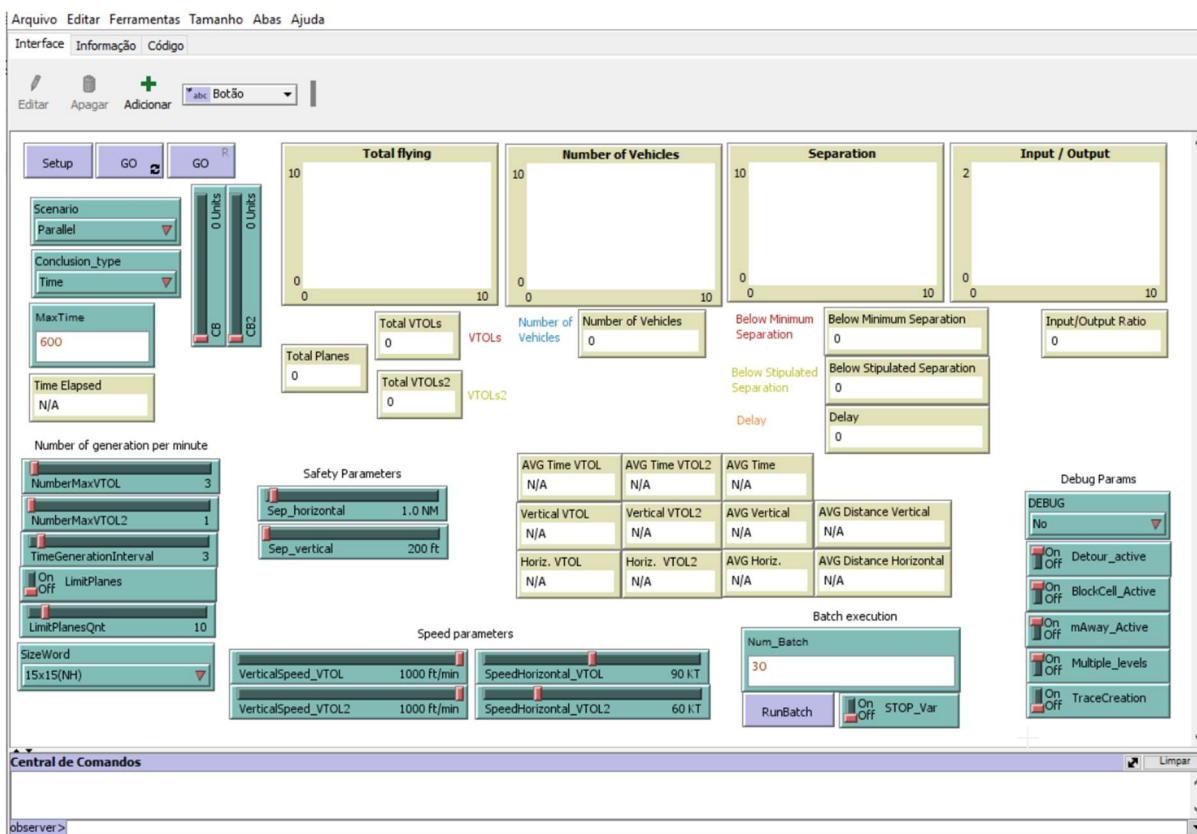


Figura 33 – Janela do Netlogo e Interface do Modelo

Fonte: Autor (2022)

Considerando-se que cada execução com determinados parâmetros produz um resultado não determinístico, em função de aspectos estocásticos existentes na execução do sistema, foi criado um modo “batch” que permite múltiplas execuções, com a utilização dos mesmos parâmetros, gerando resultados em um arquivo do tipo CSV (*Comma-Separated Values* – valores separados por vírgula). O CSV trata-se de um formato de arquivos do tipo texto em que os registros são salvos no formato texto puro, sendo que em cada linha existe um registro, e os diferentes valores ou campos dentro de uma linha são separados por vírgula. Com isso, torna-se menos trabalhosa a execução do sistema de forma múltipla, facilitando a importação dos dados gerados para uma ferramenta de análise de dados, deixando as telas do sistema com a finalidade de ser um painel de monitoramento da execução do modelo.

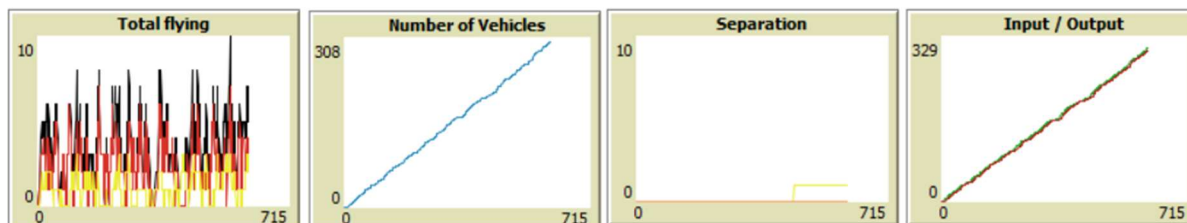


Figura 34 – Detalhes Gráficos da Interface do Modelo

Fonte: Autor (2022)

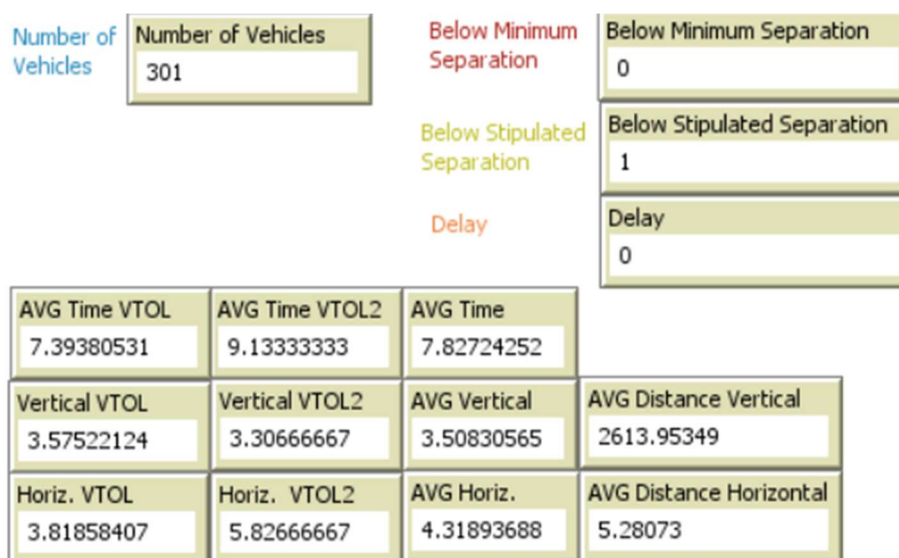


Figura 35 – Detalhe dos resultados numéricos da interface do modelo

Fonte: Autor (2022)

As execuções do modelo com os mesmos valores dos parâmetros em modo *batch* gera um arquivo no formato CSV, cujo nome explicita os valores dos parâmetros utilizados na configuração do sistema. Nesse arquivo, cada linha contém o resultado de uma execução do modelo, e cada coluna, contém os valores de uma determinada variável de saída.

Com base na Figura 33, descreve-se brevemente a seguir o Painel de Controle do sistema desenvolvido que implementa o modelo multiagente para simular o fluxo de eVTOLs em um espaço aéreo delimitado. Tem-se, no apêndice A, uma descrição detalhada dos itens que existem no modelo desenvolvido.

5.4 INTERFACE DE SIMULAÇÃO DO NETLOGO

Na interface de simulação do Netlogo, pode-se observar o ambiente em 3D de simulação, conforme janela apresada na Figura 36. Nessa figura, é mostrado o quadrilátero de simulação em bordas brancas. Esse quadrilátero corresponde ao ambiente simulado, por exemplo, de 15NMx15NM ou 30NMx30NM.

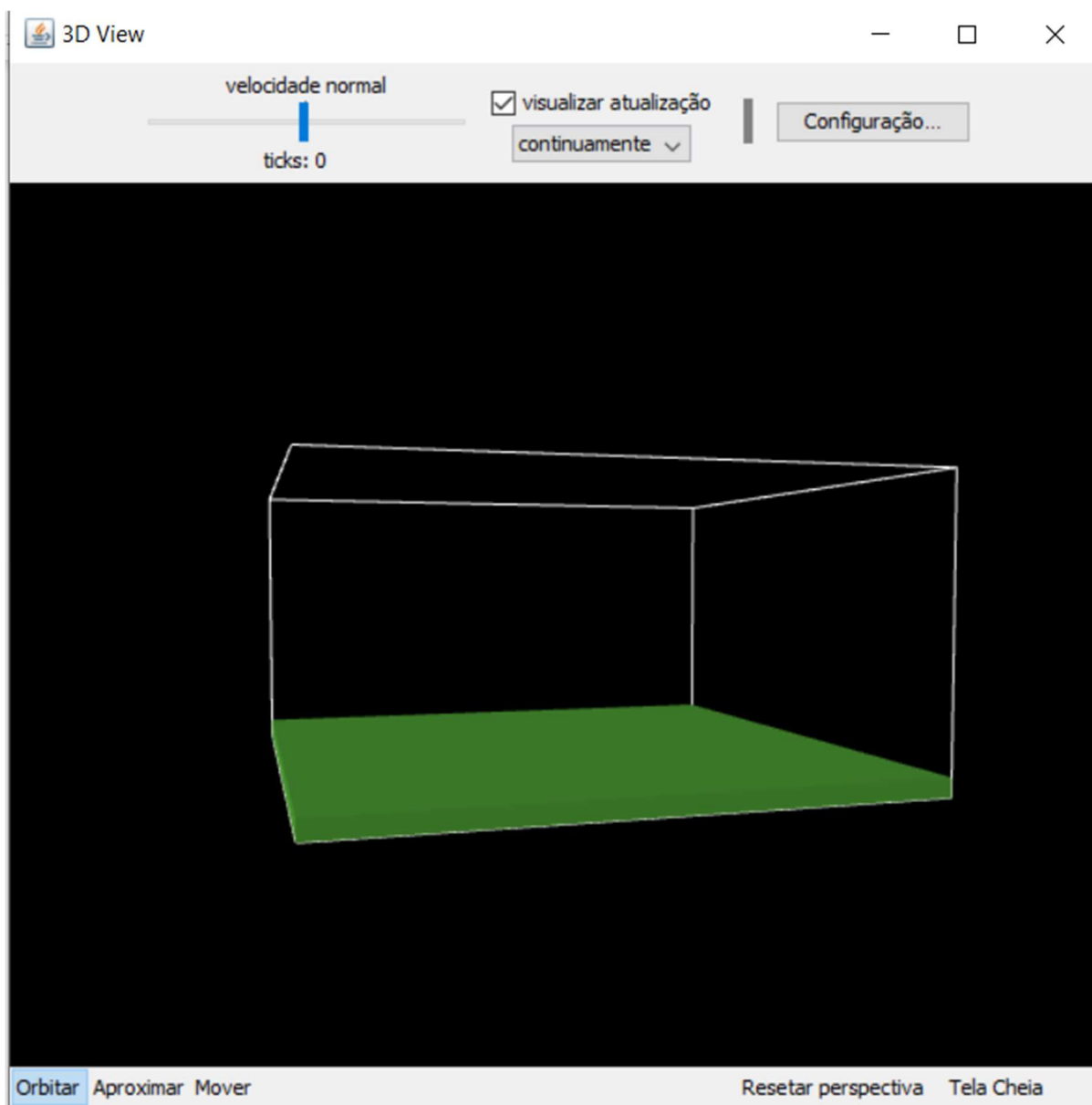


Figura 36 – Janela de simulação do Netlogo

Fonte: Autor (2022)

Na Figura 37, é mostrado o potencial de se manipular a região de simulação, podendo-se dar zoom e girar o ambiente de maneira a se navegar dentro desse

espaço. O fundo desse quadrilátero está colorido em verde, indicando a parte mais abaixo do ambiente de simulação, por onde as aeronaves entram quando decolam e saem quando pousam.

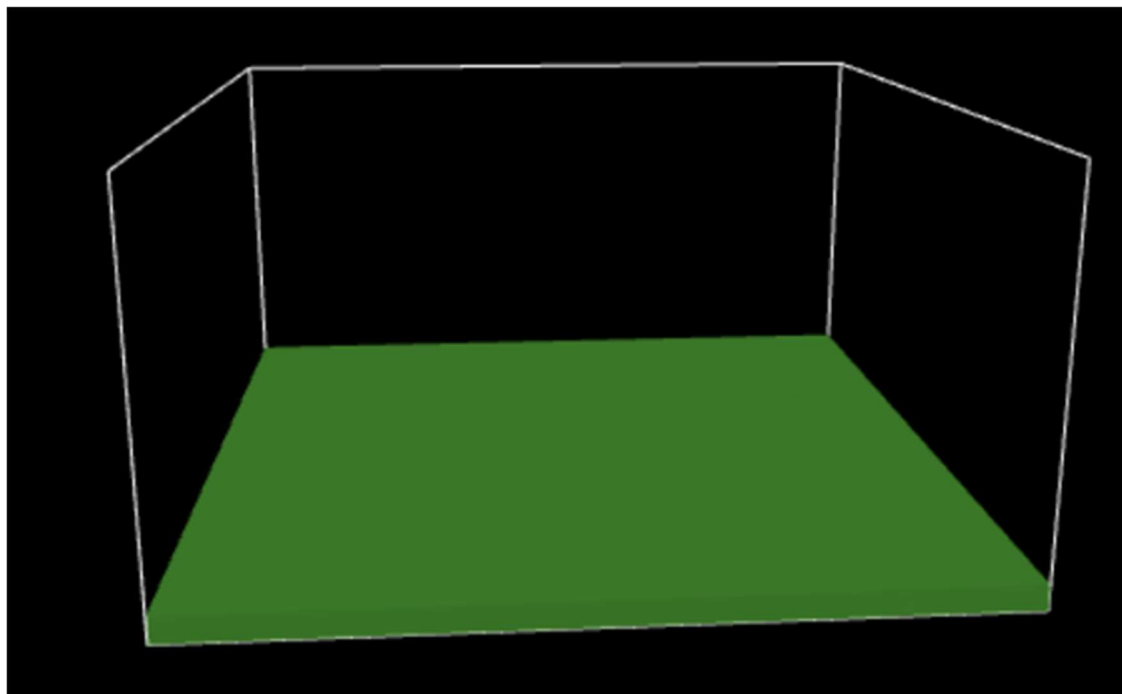


Figura 37 – Ambiente de simulação vazio (sem aeronaves)

Fonte: Autor (2022)

No Netlogo, é possível personalizar as formas de representação dos agentes, incluindo o uso de cores distintas para representar agentes diferentes. No ambiente simulado, utilizou-se a forma de um avião para ilustrar as aeronaves (no caso, VTOLs), sendo que se representaram na cor vermelha os VTOLs do tipo 1 e na cor amarela, os VTOLs do tipo 2. A Figura 38 ilustra parte do ambiente de simulação com alguns VTOLs navegando.

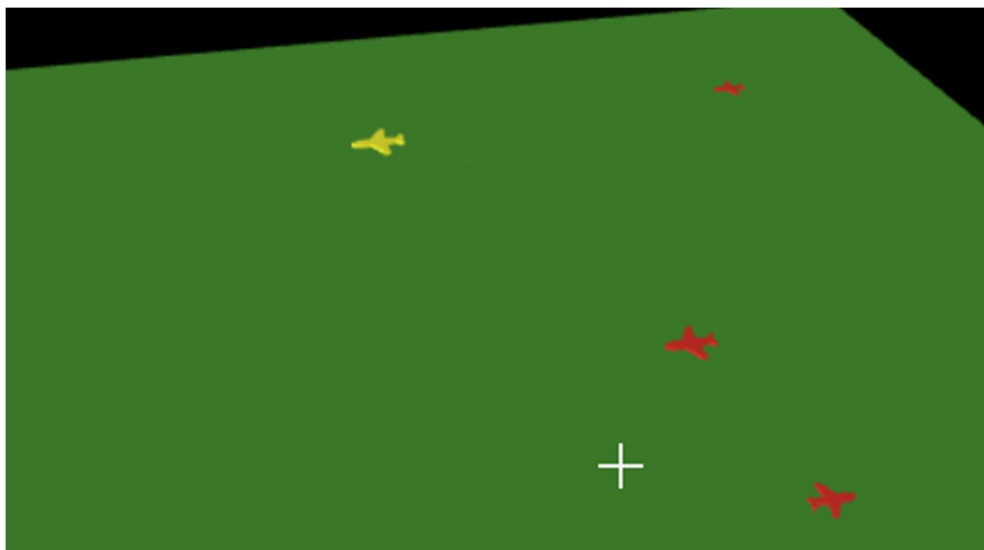


Figura 38 – Ambiente de simulação com aeronaves (agentes): em vermelho VTOLs do tipo 1 e, em amarelo, VTOLs do tipo 2

Fonte: Autor (2022)

A Figura 39 ilustra a visualização do ambiente de simulação por meio de uma vista superior, de forma que se pode ver, em 2D, a simulação que está ocorrendo em 3D. Neste caso, não se percebem os diversos níveis de altitude de navegação das aeronaves.

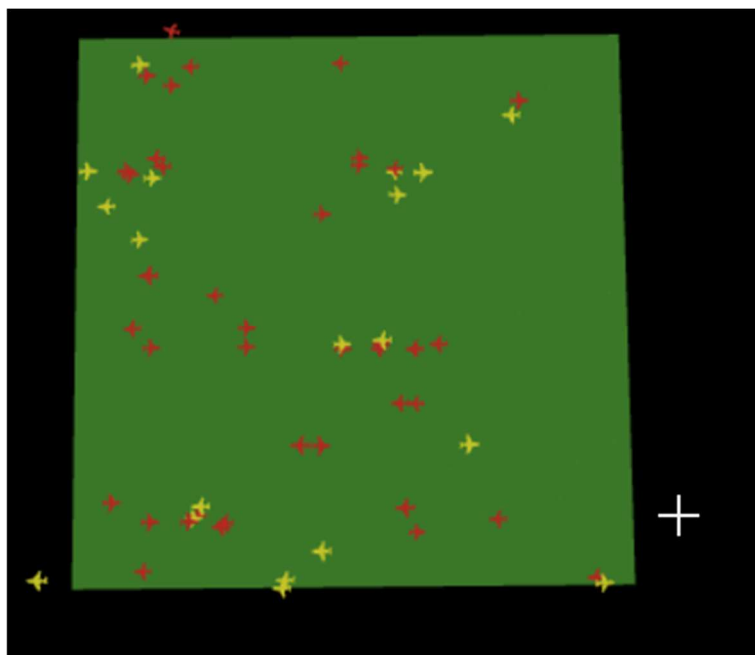


Figura 39 – Vista superior do ambiente de simulação com múltiplas aeronaves (agentes)

Fonte: Autor (2022)

Na Figura 40, pode-se visualizar essa mesma situação, porém por meio de uma vista lateral. Assim, os diversos níveis de altitude de navegação são percebidos. Com isso, o ambiente Netlogo permite uma flexibilidade na forma de se visualizar o que está sendo simulado, possibilitando a visualização da execução da simulação da melhor maneira possível.

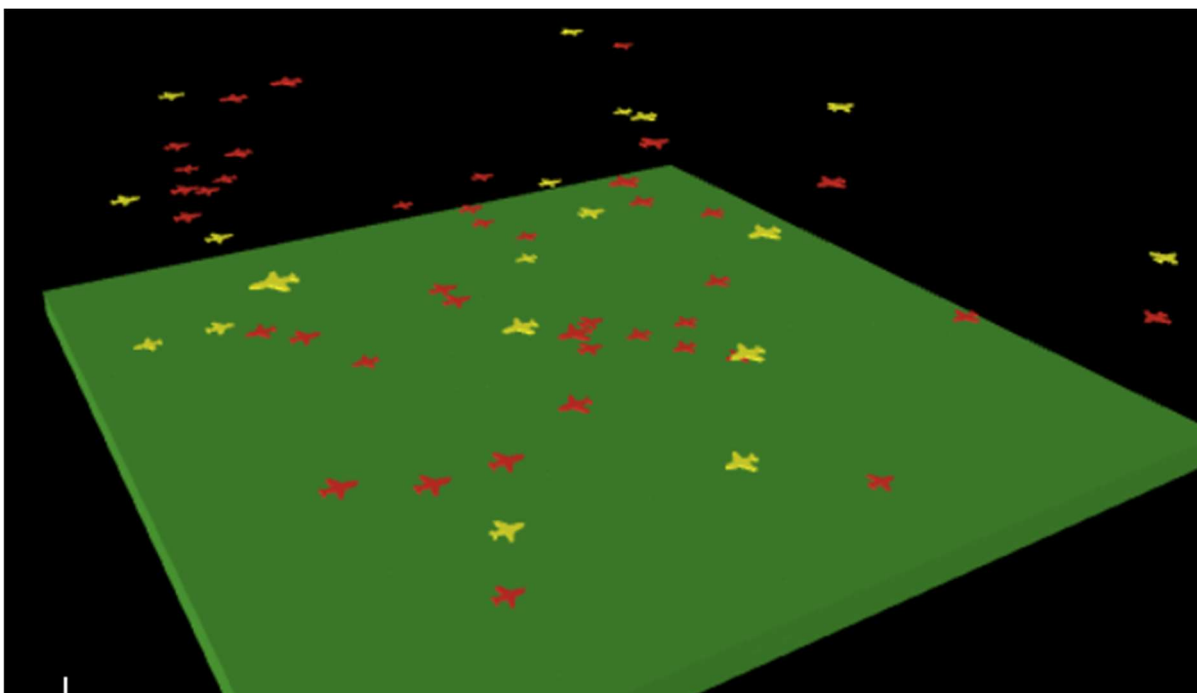


Figura 40 – Vista lateral do ambiente com agentes

Fonte: Autor (2022)

5.5 AGENTES CRIADOS NO NETLOGO

No modelo desenvolvido neste trabalho de pesquisa, definiu-se um **ciclo** como um período de tempo no qual uma quantidade estocástica (utilizando seleção de valores pseudoaleatórios com base na distribuição normal) de aeronaves é introduzida estocasticamente no espaço de simulação, sempre no início desse período de tempo, quantidade essa limitada a um valor máximo passado como parâmetro para o modelo.

Por exemplo, o período de um ciclo no modelo pode variar de 1 a 30 minutos, e o valor máximo de aeronaves (VTOLs) de cada tipo que podem entrar no ambiente a cada ciclo varia de 0 a 100 aeronaves. Dessa forma, considerando um determinado valor máximo de VTOLs que entram a cada ciclo, quanto maior o tamanho do ciclo, menos aeronaves por minuto irão poder entrar ambiente de simulação, propiciando

que as aeronaves que já estão em rota tenham mais tempo para poder se deslocar e cumprir sua missão.

Neste modelo, os eVTOLs são considerados agentes. Assim, esses agentes pousam e decolam verticalmente, e tem capacidade para se manter no ar dentro do espaço de simulação de dimensões. Na simulação desenvolvida, tanto o ponto de origem como o ponto de destino de cada uma das aeronaves encontram-se fisicamente dentro do ambiente de simulação. Assim, é atribuída uma origem escolasticamente e, dependendo do tipo de trajetória, é atribuído o destino escolasticamente, mas nunca esses pontos estarão fora da área simulada.

O modelo também considera que podem existir eVTOLs de diferentes capacidades. Dessa forma, foi considerado que podem existir eVTOLs de vários tamanhos, tipos e usos. Assim, um eVTOL de funcionalidade de ambulância ou caminhão poderia transportar grande quantidade de peso, enquanto que um VTOL de passageiros teria uma dimensão menor, e ainda, um mini-VTOL para entrega de comida e pequenas encomendas seria menor ainda. Por simplicidade, nas simulações foram considerados dois tipos de VTOLs, com diferentes velocidades (uma velocidade vertical e outra velocidade horizontal), que podem ser reguladas como parâmetros.

O modelo possui, então, como missão das aeronaves, que cada aeronave deve estar parada em qualquer ponto dos limites da simulação, devendo transportar algum passageiro ou encomenda até um outro ponto dentro desses limites, e de lá, deve esperar por outra missão, não retornando a origem.

Dentro desse modelo, os VTOLs, ao iniciarem uma viagem, possuem consciência situacional de todas as demais aeronaves em voo (que numa situação real se daria por troca de mensagens entre as aeronaves e um sistema, algo a ser realizado pelo controle do ambiente), e estas também têm as suas localizações conhecidas. Os agentes do tipo aeronave não compartilham entre si suas trajetórias nem velocidades, mas somente suas localizações (que correspondem a cada quadrícula ocupada por cada aeronave), já que não existe um sistema centralizado para controle de aeronaves. A exceção corresponde ao sistema centralizado de controle das condições climáticas, que compartilha, com todas as aeronaves, as posições dos CBs.

Ainda neste modelo, as quadrículas também são modeladas como agentes. Elas podem informar a agentes aeronaves se nelas existem formações de nuvens (CBs) e, por isso, não são navegáveis e devem ser evitadas. Quando estão ocupadas

por CBs, as quadrículas ficam bloqueadas para a navegação. Ainda, antes de uma aeronave iniciar o processo de subida ou descida, ela solicita o bloqueio de uma quadrícula e, posteriormente, o seu desbloqueio. As aeronaves podem, a qualquer momento, inquirir qualquer quadrícula para saber o seu estado atual (bloqueada por um CB, bloqueada por outra aeronave ou desbloqueada) para, então tomar suas ações. Esse mecanismo visa simular um sistema centralizado capaz de efetuar bloqueio de áreas e prover informações de clima.

As aeronaves tentam desenvolver uma estratégia gulosa individualmente (alcançar o seu destino da melhor maneira possível considerando), sem considerar possíveis casos em que uma otimização ou colaboração com as demais aeronaves poderiam gerar rotas mais otimizadas. Uma estratégia gulosa consiste em um agente tentar obter o resultado mais otimizado, individualmente, sem se importar com a otimização dos resultados dos outros agentes. Assim, neste caso, o agente “aeronave” tenta obter um trajeto que seja o mais rápido possível sem se importar com os outros agentes, o que pode fazer com que outro veículo tenha que esperar no ar e, assim, não vai existir uma otimização do tempo global de traslado de todas as aeronaves, já que não existe um agente controlador centralizado de trajetórias. Esse cenário é basicamente o mesmo cenário que existe hoje envolvendo veículos terrestres, no qual os motoristas tentam chegar aos seus destinos buscando otimizar somente o seu trajeto. Apesar disso, cada aeronave leva em consideração as ações e movimentações de outras aeronaves, na tentativa de executar as ações de maneira segura, sem efetuar violações relacionadas com a segurança da operação (riscos de colisão).

As aeronaves consultam as quadriculas para saber se existe um dos tipos de bloqueio nelas e, caso elas estejam bloqueadas, as aeronaves planejam um desvio para poder passar ou esperar o espaço em volta ficar liberado. O fato de a aeronave ir liberando o espaço durante a subida e descida logo após terminar de passar facilita manter o espaço fechado pelo menor espaço de tempo possível. O objetivo é manter o espaço aéreo o mais auto-gerenciado possível, e de maneira descentralizada, sem a necessidade de um controlador ou gerenciador humano do espaço aéreo. O único aspecto centralizado seria um sistema que gerenciaria os bloqueios de quadriculas, efetuados ou por aeronaves, ou por questões adversas do clima.

5.6 INTERFACE DOS AGENTES NO NETLOGO

No modelo multiagente desenvolvido foram criados dois tipos de agentes, chamados de VTOL-1 e VTOL-2, que têm as mesmas funcionalidades e regras no modelo, mas que diferem entre si por possuírem parâmetros diferentes. Isso se deve ao fato de terem sido considerados neste modelo dois tipos de aeronaves: uma para o transporte de cargas e outra para o transporte de pessoas. Esses dois tipos de VTOLs possuem tamanhos, pesos e parâmetros de desempenho (como velocidades horizontal e vertical) diferenciados. Parâmetros como autonomia da aeronave não foram considerados, pois não fazem parte do escopo de trabalho.

O código desenvolvido na linguagem Netlogo utiliza uma versão multiagente da linguagem Logo. No código escrito em Netlogo, é possível criar variáveis globais de ambiente, definir estruturas de dados dentro de agentes (como aeronaves e quadrículas) e definir códigos de ação para serem executados pelos agentes ou pelo ambiente.

Todo o código desenvolvido para este trabalho de pesquisa encontra-se no Apêndice A. Por exemplo, na Figura 41, apresenta-se a estrutura de dados concebida para o agente “aeronave” (UASs-own), ilustrando todas as variáveis criadas para este tipo de agente no modelo.

```

UASs-own [
  destX destY destZ          ;; Destiny cordinates
  OrigX OrigY OrigZ        ;; Original Location
  colision                  ;; Colision Flag
  angV                      ;; AngV is used when the diagonal running
  head                      ;; saves the head for a manovering
  stored_pitch              ;;
  x_last y_last z_last last_count ;; save the last cordinates of the plane
  dist                     ;; distance flyed cont
  dist_H dist_V            ;; distance moved vertically and Horizontaly
  time time_V time_H H_flag V_flag ;; time the plane used to get to the target
  X_org                    ;; manovering flag
  Y_org                    ;; manovering flag
  desvioA desvioB desvioC  ;; manovering flags
]

```

Figura 41 – Estrutura de Dados do Agente “Aeronave”

Fonte: Autor (2022).

Já na Figura 42, observam-se as várias variáveis que existem na estrutura de código de cada quadrícula (patches-own) que existe no ambiente do Netlogo.

```

patches-own [
  bloqueado ;; blocked flag
  quem      ;; who blocked the patch
  quando    ;; when the patch get blocked
  cb_state  ;; if the patch is a CB flag
]

```

Figura 42 – Estrutura de Dados da “Quadrícula”

Fonte: Autor (2022)

Na Figura 43, encontra-se o código executado no nível de ambiente, que cria novos tráfegos a cada ciclo de execução da simulação, seguindo diversas configurações do ambiente e valores de variáveis configuradas no Painel de Controle. Em suma, esse código define quanto tráfego deve ser criado.

```

to GenerateLoad
  if(DEBUG = "No") [
    ifelse(TimeGeneration > (TimeGenerationInterval - 1)) [
      ifelse(LimitPlanes) [
        ifelse((NumberMaxVTOL + NumberMaxVTOL2 + (count turtles)) >= LimitPlanesQnt) [
          let diff ((NumberMaxVTOL + NumberMaxVTOL2 + (count turtles)) - LimitPlanesQnt)
          let diff_1 ceiling (diff / 2)
          if(NumberMaxVTOL - diff_1 > 0) [GenerateVTOL NumberMaxVTOL - diff_1 - 1]
          if(NumberMaxVTOL2 - diff_1 > 0) [GenerateUAV NumberMaxVTOL2 - diff_1 - 1]
        ]
        [
          GenerateVTOL NumberMaxVTOL
          GenerateUAV NumberMaxVTOL2
        ]
      ]
      [
        GenerateVTOL NumberMaxVTOL
        GenerateUAV NumberMaxVTOL2
      ]
    ]
    set TimeGeneration 0
  ]
  [
    set TimeGeneration (TimeGeneration + 1)
  ]
  set_txInput
]
end

```

Figura 43 – Exemplo de código de geração de agentes

Fonte: Autor (2022)

5.7 CUMULONIMBUS (CBs)

No ambiente de simulação do Netlogo foram criados os chamados Cumulonimbus (CBs), que representam condições climáticas adversas, modelando áreas do espaço aéreo que apresentam condições climáticas desfavoráveis para a

navegação aérea, e que devem, portanto, ser bloqueadas para o tráfego, por questão de segurança. Na prática, os CBs correspondem a formações de nuvens densas, ventos fortes ou chuvas fortes.

Isso foi implementado no modelo por meio da construção de dois tipos de CBs (CB-1 e CB-2), modelando duas características climáticas consistentes com diferentes tipos de formações climáticas que podem ocorrer. Cada CB causa o bloqueio de 4 quadrículas (cada quadrícula modelada possui 0,5NM x 05NM na horizontal por 200ft de altura). Essas diferentes dimensões de formações climáticas correspondem a diferentes formatos de CBs existentes. Para isso, foi utilizado um tamanho de 4 quadrículas para os CBs, considerando que um CB de tempestade de dimensão menor do que isso seria muito pequeno para efetivamente perturbar o tráfego de aeronaves, enquanto que um CB muito maior do que isso seria pouco provável de existir. É possível escolher o número de CBs de cada tipo que serão colocados no modelo em cada simulação realizada.

Na Figura 44, têm-se as ilustrações de como foram modelados cada tipo de CB. O CB-1 é composto de 4 quadrículas na vertical (sendo assim, 0,5 NM x 0,5 NM de área por 800 ft. de altura), enquanto que o CB-2 é composto de 2 colunas com 2 quadrículas de altura cada (sendo assim, 1 NM x 1 NM de área por 400 ft. de altura). Assim, o CB-1 representa fenômenos climáticos que ficam localizados em uma área menor, porém afetam uma grande região em questão de altitude, enquanto que o CB-2 representa fenômenos climáticos que não têm tanta altitude, mas que se afetam uma área maior do espaço aéreo.

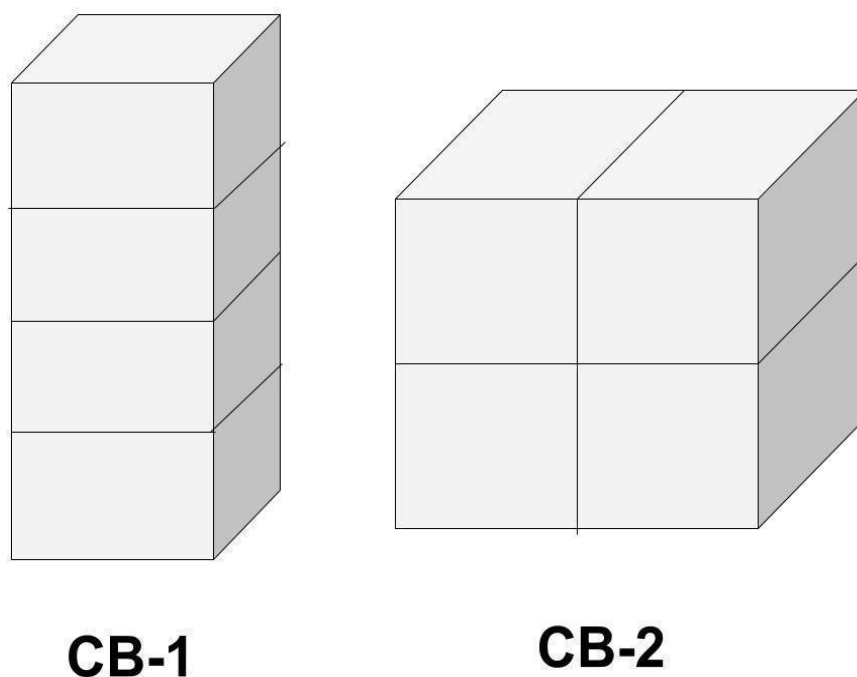


Figura 44 – Diagrama dos diferentes tipos de CBs (CB-1 e CB-2)

Fonte: Autor (2022).

O sistema multiagente desenvolvido, no início de cada simulação, verifica os parâmetros correspondentes aos números de CBs selecionados de cada tipo e gera, aleatoriamente no espaço de simulação, a cada execução, o número de CBs especificado. Foi considerado neste modelo que esses CBs duram e permanecem no mesmo lugar durante todo o período da simulação.



Figura 45 – CB Comprida ao lado de uma aeronave

Fonte: Minhapaixaoeaviaoaviacao (2023)



Figura 46 – CB Comprida ao lado de uma aeronave

Fonte: rocket cumulonimbus (2023)

Podemos ver na Figura 45, um exemplo de uma CB extremamente comprida próxima a uma aeronave, e na Figura 46 vemos uma figura com uma CB que não possui muita altura, mas se estende por uma grande área tirada do solo. Esses padrões de CBs que foram modelados nesse trabalho, como veremos a seguir.

Esses tipos de formações climáticas podem ser observados no ambiente de simulação por meio das figuras a seguir. Primeiramente, na Figura 47, tem-se um exemplo de uma configuração do ambiente com alguns CBs do tipo 1 (CB-1). Já na Figura 48, há uma configuração com alguns CBs do tipo 2 (CB-2).

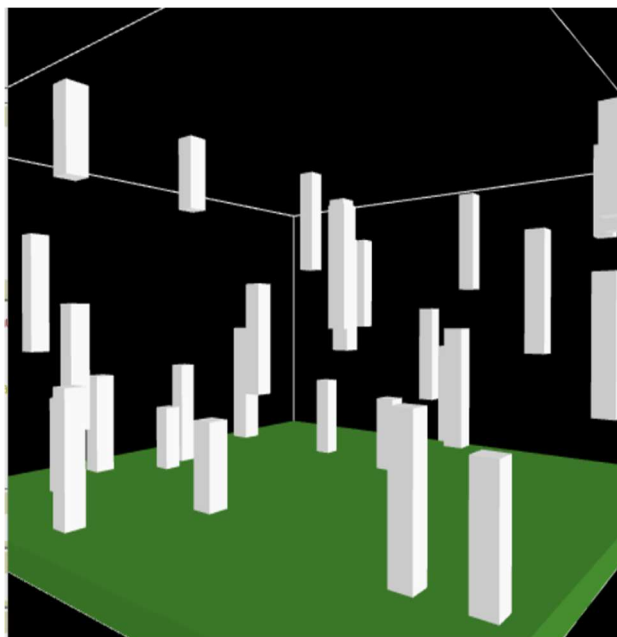


Figura 47 – CB-1 no ambiente de simulação
Fonte: Autor (2022)

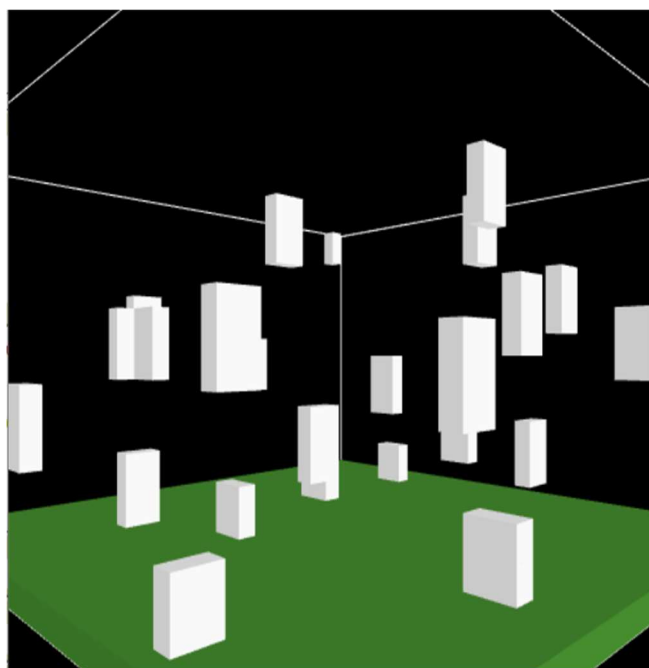


Figura 48 – CB-2 no ambiente de simulação
Fonte: Autor (2022)

O modelo multiagente desenvolvido, ao sortear de modo estocástico a posição de um CB qualquer, o faz colocando o seu centro na posição sorteada. Por isso, em alguns casos, pode-se ter alguns CBs menores em tamanho que os demais, e isso ocorre somente quando parte deles ficarem fora do espaço simulado (nas bordas do espaço simulado). Outro ponto importante de se notar é que é possível também criar

simulações usando os dois tipos de CBs simultaneamente no mesmo ambiente de simulação.

5.8 CENÁRIOS DE TRAJETÓRIAS DE AERONAVES

Neste trabalho, as aeronaves podem assumir alguns tipos de trajetórias, que são a partir de agora denominadas de cenários. Cada aeronave pode assumir três tipos de cenários para a geração de trajetórias, as quais serão chamadas de **paralelo**, **perpendicular** e **geral**, e que são descritas a seguir.

1- Cenário Paralelo:

A Figura 49 ilustra um cenário simples com 4 aeronaves (A), sempre partindo de uma origem (O, em verde) rumo a um destino (X, em vermelho), sendo que as trajetórias são indicadas por linhas tracejadas e a seta da linha indica a direção seguida pela aeronave. Esta ilustração mostra uma visualização das trajetórias “por cima”, e todas as trajetórias entre origem e destino seguem a mesma direção na horizontal.

Eventuais conflitos de trajetória ocorrerão, neste cenário, apenas na parte da frente ou na parte de trás de uma aeronave.

Nesse exemplo, uma das aeronaves necessita ultrapassar outra aeronave mais lenta que se encontra na mesma direção e sentido e no mesmo nível de voo; nesta situação, a aeronave mais rápida realiza um desvio pela direita, sempre para uma quadrícula mais à direita, retornando depois a mesma direção e sentido original. Uma vez ultrapassada a aeronave mais lenta, a aeronave mais rápida volta para a sua trajetória original, caminhando normalmente para o seu destino.

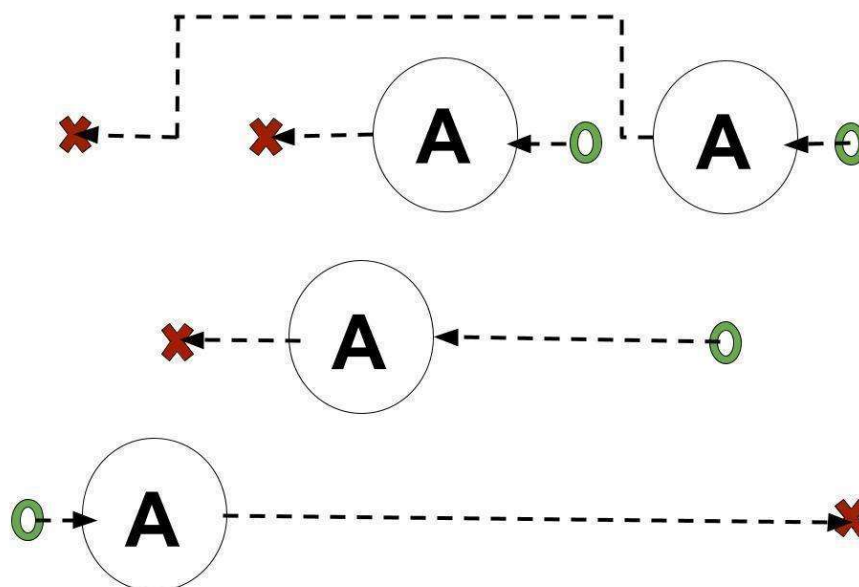


Figura 9 – Esquema de cenário com trajetórias paralelas e ultrapassagem pela direita

Fonte: Autor (2022)

2- Cenário Perpendicular:

A Figura 50 ilustra um cenário com 5 aeronaves (A), que apresentam trajetórias, no plano, na horizontal e na vertical, ou seja, essas trajetórias podem se cruzar perpendicularmente, partindo de uma origem (O, em verde) rumo a um destino (X, em vermelho), sendo que as trajetórias referentes a aeronaves com deslocamento horizontal são indicadas por linhas tracejadas pretas, enquanto que as trajetórias referentes a aeronaves com deslocamento vertical são indicadas por linhas tracejadas azuis. Ainda, a seta da linha indica a direção seguida pela aeronave.

Eventuais conflitos de trajetória ocorrerão, neste cenário, em rotas em uma mesma direção ou em rotas que se cruzam de forma perpendicular, considerando sempre aeronaves navegando em um mesmo nível.

Nesse exemplo, uma das aeronaves necessita ultrapassar outra aeronave mais lenta que se encontra na mesma direção e sentido e no mesmo nível de voo; nesta situação, a aeronave mais rápida realiza um desvio pela direita, sempre para uma quadrícula mais à direita, retornando depois à mesma direção e

sentido original; uma vez ultrapassada a aeronave mais lenta, a aeronave mais rápida volta para a sua trajetória original, caminhando normalmente para o seu destino. Deve-se destacar que, além de realizar a ultrapassagem da aeronave mais lenta, esta aeronave deve se preocupar com um eventual conflito com outra aeronave trafegando na vertical que cruza a sua trajetória e, caso necessário, deve ficar “parada”, aguardando a outra aeronave na vertical passar.

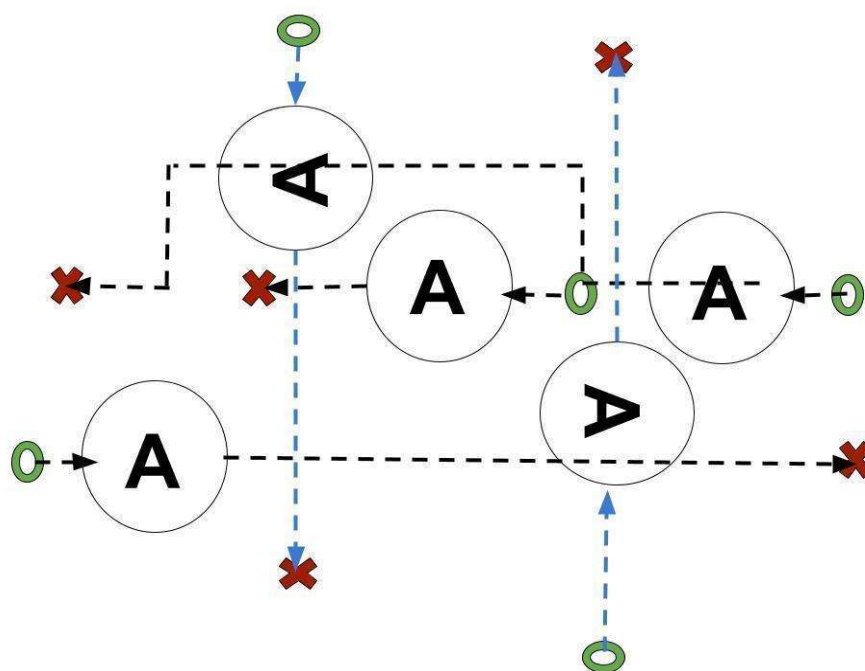


Figura 50 – Esquema de cenário com trajetórias perpendiculares e ultrapassagem pela direita
Fonte: Autor (2022)

3- Cenário Geral:

A Figura 51 ilustra um cenário com 5 aeronaves (A), que apresentam trajetórias, no plano, na horizontal, na vertical e de forma perpendicular, ou seja, essas trajetórias podem se cruzar perpendicular ou diagonalmente, partindo de uma origem (O, em verde) rumo a um destino (X, em vermelho), sendo que as trajetórias referentes a aeronaves com deslocamento horizontal são indicadas por linhas tracejadas pretas, as trajetórias referentes a aeronaves com deslocamento vertical são indicadas por linhas tracejadas azuis e as trajetórias referentes a aeronaves com deslocamento diagonal são indicadas por linhas

tracejadas amarelas. Ainda, a ponta da seta indica a direção seguida pela aeronave.

Eventuais conflitos de trajetória ocorrerão, neste cenário, em rotas em uma mesma direção ou em rotas que se cruzam, de forma perpendicular ou diagonal, considerando sempre aeronaves navegando em um mesmo nível.

Nesse exemplo, uma das aeronaves necessita ultrapassar outra aeronave mais lenta que se encontra na mesma direção e sentido e no mesmo nível de voo; nesta situação, a aeronave mais rápida realiza um desvio pela direita, sempre para uma quadrícula mais à direita, retornando depois à mesma direção e sentido original; uma vez ultrapassada a aeronave mais lenta, a aeronave mais rápida volta para a sua trajetória original, caminhando normalmente para o seu destino. Deve-se destacar que, além de realizar a ultrapassagem da aeronave mais lenta, esta aeronave deve se preocupar com eventuais conflitos com outras aeronaves trafegando na vertical ou na diagonal, que cruzam a sua trajetória e, caso necessário, deve ficar “parada”, aguardando a(s) outra(s) aeronave(s) na vertical ou na diagonal passar.

Este tipo de cenário foi chamado de “geral”, pois é o que mais se aproxima de uma situação real de espaço aéreo congestionado.

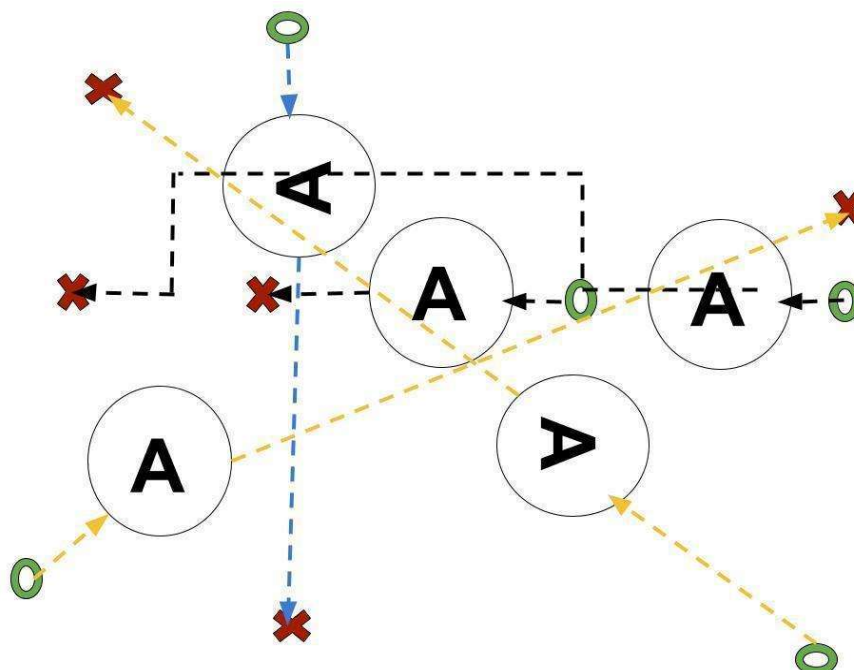


Figura 51 – Esquema de cenário com trajetórias gerais e ultrapassagem pela direita

Fonte: Autor (2022)

5.9 PARÂMETROS DO MODELO MULTIAGENTE

Como já apresentado anteriormente, o modelo é alimentado com uma série de parâmetros de entrada e, então, produz uma série de resultados. Como parâmetros de entrada para o modelo, têm-se os seguintes:

- 1- Tipo de cenário: pode ser do tipo Paralelo, Perpendicular ou Geral;
- 2- Tipo de Conclusão: o término da execução da simulação pode se dar por: a) tempo (execução por um intervalo de tempo determinado); b) ocorrência de um acidente; c) ocorrência de um conflito; d) ocorrência de um *delay*, quando uma aeronave permanece em estado de espera no espaço aéreo por mais de 10 minutos;
- 3- Tempo máximo de execução: representa o tempo máximo de execução da simulação, em minutos, caso a finalização da simulação por tempo esteja configurada (item 2a);
- 4- Número de CBs (Tipo 1): representa o número de CB-1 inseridos no espaço de simulação, que varia entre 0 e 100. Zero nuvens equivalem a não existir um CB no ambiente de simulação, enquanto que 100 CBs ocupariam mais espaço do que de fato existe no ambiente de simulação;
- 5- Número de CBs (Tipo 2): representa o número de CB-2 inseridos no espaço de simulação, que varia entre 0 e 100. Zero nuvens equivalem a não existir um CB no ambiente de simulação, enquanto que 100 CBs ocupariam mais espaço do que de fato existe no ambiente de simulação;
- 6- Número máximo de VTOLs (Tipo 1): representa o número máximo de VTOL-1 que entra na simulação a cada ciclo, sendo um valor numérico que varia entre 0 e 100. Mais do que 100 VTOLs em um ciclo não faria sentido prático, além de deixar o ambiente de simulação muito sobrecarregado, podendo causar congestionamentos difíceis de serem administrados;
- 7- Número máximo de VTOLs (Tipo 2): representa o número máximo de VTOL-2 que entra na simulação a cada ciclo, sendo um valor numérico que varia entre 0 e 100. Mais do que 100 VTOLs em um ciclo não faria sentido prático,

além de deixar o ambiente de simulação muito sobrecarregado, podendo causar congestionamentos difíceis de serem administrados;

- 8- Tempo de geração de VTOLs: define quanto tempo vai ser o ciclo em que vão ser geradas novas aeronaves dentro do modelo (em minutos), sendo um valor numérico que varia entre 1 e 30. Mais do que 30 minutos para um ciclo não tem sentido prático, pois deixaria o ambiente muito ocioso;
- 9- Tamanho do mundo: define o tamanho das distâncias na horizontal do ambiente de simulação, podendo ser 15x15 ou 30x30 (em MN);
- 10- Separação de segurança vertical: indica a distância de separação mínima na vertical que as aeronaves têm que manter para não gerar um aviso de conflito, podendo assumir valores entre 0,5 NM e 10 NM). Menos do que 0,5 NM não faz sentido prático, pois seria menor do que a dimensão horizontal de uma quadrícula, enquanto que mais do que 10 NM é uma dimensão quase do tamanho do ambiente de simulação;
- 11- Separação de segurança horizontal: indica a distância de separação mínima na horizontal que as aeronaves têm que manter para não gerar um aviso de conflito, podendo assumir valores entre 200 ft e 1.000 ft. Menos do que 200 ft não faz sentido prático, pois seria menor do que a dimensão vertical de uma quadrícula, enquanto que mais do que 1.000 ft é uma dimensão quase da altura do ambiente de simulação;
- 12- Velocidade vertical (VTOL Tipo 1): indica a velocidade do VTOL-1 quando ele se desloca verticalmente, representada em ft/min (pés por minuto), variando entre 200 e 1.000, aumentando de 200 em 200 ft. A velocidade e o aumento da velocidade vertical estão relacionados com o tamanho da quadrícula, para no mínimo a aeronave se deslocar de uma quadrícula para outra no espaço de tempo de 1 minuto;
- 13- Velocidade vertical (VTOL Tipo 2): indica a velocidade do VTOL-2 quando ele se desloca verticalmente, representada em ft/min (pés por minuto), variando entre 200 e 1.000, aumentando de 200 em 200 ft. A velocidade e o aumento da velocidade vertical estão relacionados com o tamanho da quadrícula, para no mínimo a aeronave se deslocar de uma quadrícula para outra no espaço de tempo de 1 minuto;

- 14- Velocidade horizontal (VTOL Tipo 1): indica a velocidade do VTOL-1 quando ele se desloca horizontalmente, representada em nós (*knot* – KT), variando entre 30 e 150, aumentando de 30 em 30 nós. A velocidade e o aumento da velocidade horizontal estão relacionados com o tamanho da quadrícula, para no mínimo a aeronave se deslocar de uma quadrícula para outra no espaço de tempo de 1 minuto;
- 15- Velocidade horizontal (VTOL Tipo 2): indica a velocidade do VTOL-2 quando ele se desloca horizontalmente, representada em nós (*knot* – KT), variando entre 30 e 150, aumentando de 30 em 30 nós. A velocidade e o aumento da velocidade horizontal estão relacionados com o tamanho da quadrícula, para no mínimo a aeronave se deslocar de uma quadrícula para outra no espaço de tempo de 1 minuto

Como resultados produzidos mediante a execução da simulação do modelo, tem-se como saída o seguinte:

- 1- Tempo de execução da simulação: representa o tempo simulado durante a execução do modelo (em minutos). Não é o tempo real que a simulação demorou para ser executada;
- 2- Número total de aeronaves (Total / VTOL-1 / VTOL-2): representa o número de aeronaves de cada tipo e total que continuam no modelo;
- 3- Número de veículos que completaram o trajeto: representa o número total de aeronaves que completaram seus trajetos, saindo do ponto original de decolagem e atingindo o ponto de destino para pouso;
- 4- Número total de aeronaves abaixo da separação mínima: representa o número total de aeronaves que, durante todo o período de simulação, ultrapassaram o limite de segurança de separação vertical ou horizontal mínima (vide seção 5.10 para maiores detalhes);
- 5- Número total de aeronaves abaixo da separação estipulada: representa o número total de aeronaves que, durante todo o período de simulação, ultrapassaram o limite de estipulado da separação vertical ou horizontal

mínima, conforme definido nos parâmetros da simulação (vide seção 5.10 para maiores detalhes);

- 6- Número de aeronaves bloqueadas (com *delay*): representa o número total de aeronaves no estado de espera (em *delay*);
- 7- Tempo médio de trajetórias: representa o tempo médio (em minutos) para a aeronave completar a trajetória, correspondendo à média dos tempos de percurso de todas as aeronaves que terminaram sua trajetória com sucesso, desde a sua decolagem até o seu pouso;
- 8- Tempo médio de trajetórias (vertical): representa o tempo médio (em minutos) de deslocamento vertical de todas as aeronaves que terminaram sua trajetória com sucesso;
- 9- Tempo médio de trajetórias (horizontais): representa o tempo médio (em minutos) de deslocamento horizontal de todas as aeronaves que terminaram sua trajetória com sucesso;
- 10- Distância média percorrida (vertical): representa a distância média (em pés) do deslocamento vertical de todas as aeronaves que terminaram sua trajetória com sucesso;
- 11- Distância média percorrida (horizontal): representa a distância média (em milhas náuticas) do deslocamento horizontal de todas as aeronaves que terminaram sua trajetória com sucesso.

5.10 SEPARAÇÃO MÍNIMA E SEPARAÇÃO ESTIMULADA

Existe uma diferenciação no modelo de simulação entre os conceitos de "**separação de segurança**" (referenciada neste trabalho como "separação mínima") e de "**separação definida**" (referenciada neste trabalho como "separação estipulada"), conforme ilustrado na Figura 52.

Nessa figura, a **separação de segurança** é mostrada em (1), definida por uma quadrícula de dimensões 200ft por 0.5 NM, sendo esse o menor espaço considerado para uma aeronave ocupar. Então, caso duas aeronaves se encontrem nesse mesmo espaço, considera-se que houve uma violação dessa separação de segurança e, portanto, que ocorreu um acidente (colisão) ou quase acidente. Ainda nessa mesma

figura, ilustra-se o espaço de **separação definida ou estipulada**, como mostrado em (2), que corresponde ao espaço de segurança que deve ser respeitado para evitar um possível acidente (colisão entre aeronaves). Esse valor de distância pode ser configurado como parâmetro de entrada para o modelo, devendo as dimensões desse espaço serem, no mínimo, as dimensões de uma quadrícula, mas pode ser maior. Obviamente o aumento desse parâmetro irá diretamente impactar na capacidade do espaço aéreo total (espaço de simulação), representado por (3), já que deve existir alguma margem de manobra para as outras aeronaves se movimentarem.

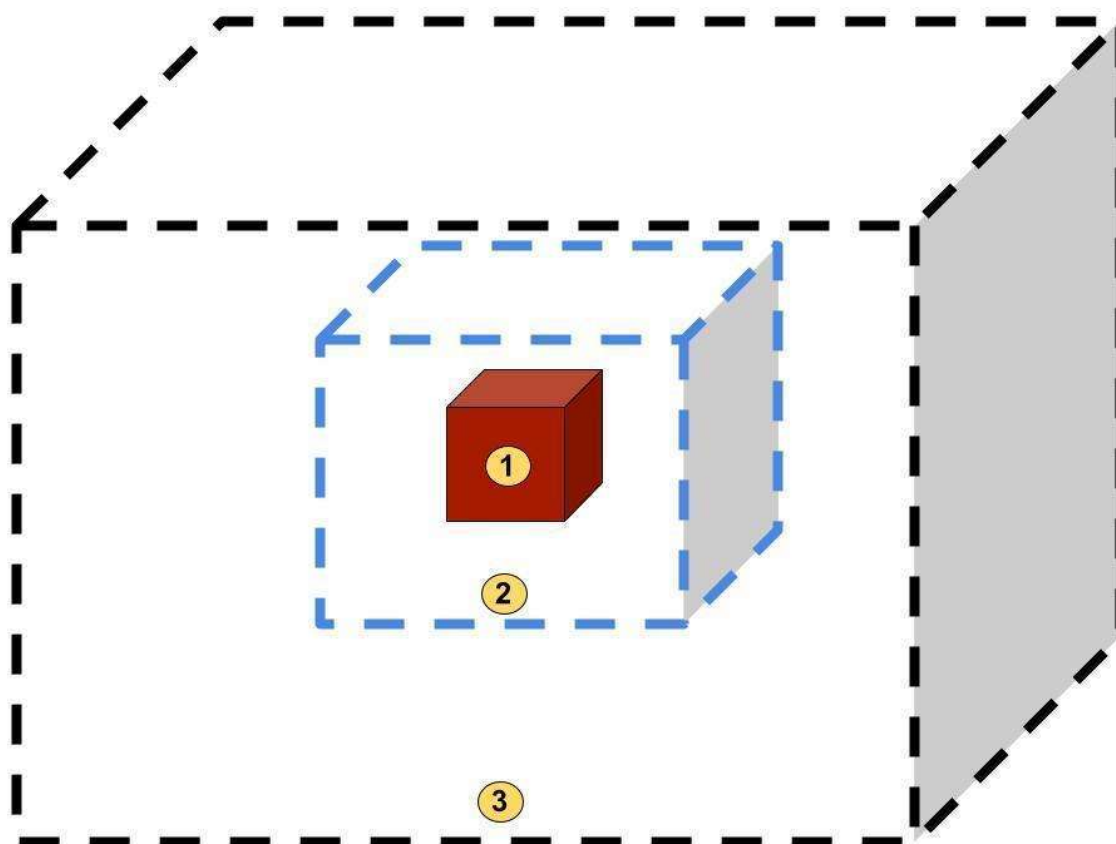


Figura 52 – Áreas de separação: (1) separação de segurança; (2) separação definida; (3) espaço de simulação

Fonte: Autor (2022)

5.11 ESTADO DE ESPERA DE AERONAVES (ESTADO DE *DELAY*)

Um parâmetro importante definido no modelo de simulação corresponde ao estado de espera de uma aeronave, quando a aeronave deve aguardar, parada no ar, a quadrícula seguinte que ela deseja ocupar ser desbloqueada por outra aeronave.

Esse fato ocorre, principalmente, quando o espaço aéreo está congestionando, gerando muitas trajetórias conflitantes.

Nos casos de sobrecarga do espaço aéreo pode-se ter múltiplas aeronaves em condições de espera grandes. Dessa forma, decidiu-se colocar um alerta no sistema de simulação, que sinaliza quando uma aeronave está parada na mesma quadrícula por 10 minutos; neste caso, o sistema automaticamente muda o estado dessa aeronave para "atrasada" (*delay*). Com isso, é possível contar o número de aeronaves no ambiente nesse estado, além de se visualizar os parâmetros de desempenho médio do sistema. Assim, é possível se prever colapsos do sistema, detectar a sobrecarga do sistema com muitas aeronaves voando ou detectar congestionamentos localizados. Essa variável é importante para detectar se o ambiente está lidando bem com a carga de aeronaves durante a simulação.

O modelo multiagente tenta garantir que não existam violações dos parâmetros de segurança, evitando colisões. Mas caso o número de aeronaves entrando no sistema apresente um padrão de entrada de aeronaves muito maior do que de saída de aeronaves, serão gerados congestionamentos e problemas decorrentes da superlotação do espaço aéreo. O modelo não bloqueia, no seu modo padrão, o número de aeronaves que estão entrando; porém, foi incluída uma opção para limitar o número de aeronaves, com o objetivo de se encontrar situações em que se atinja a capacidade desse modelo.

5.12 DEFINIÇÃO DO MODELO PELO PROTOCOLO ODD

O ODD (*Overview, Design Concepts, Details*) é um protocolo usado para a descrição de sistemas baseados em agentes. Seu nome advém de suas 3 categorias: *Overview* (visão geral), *Design Concepts* (conceitos de *design*) e *Details* (detalhes). A primeira categoria presta-se a uma descrição geral do sistema; a segunda explicita como os agentes se comportam; e a última coloca alguns detalhes sobre o funcionamento do modelo (Grimm et al., 2020). Nas tabelas a seguir, tem-se uma descrição do modelo do sistema multiagente proposto neste trabalho, sendo que esta descrição está baseada no protocolo ODD relatado em Grimm et al. (2020), que permite descrever de maneira sistemática os principais pontos e pressupostos utilizados na modelagem.

Utiliza-se, na Tabela 2, uma descrição simplificada do modelo construído neste trabalho baseada em Grimm et al. (2020). Já a Tabela 3 contém uma recomendação proposta em Grimm et al. (2020), do modelo ODD completo, dividindo as 3 categorias em 7 subcategorias: as 3 primeiras subcategorias encontram-se atreladas à primeira categoria; a quarta subcategoria está inclusa na segunda categoria, e as três últimas subcategorias pertencem à terceira e última categoria (Grimm et al., 2020).

Na subcategoria “Propósito e padrões” é colocado, de maneira geral, o propósito do modelo e o que é modelado. Na segunda subcategoria, “Entidades, variáveis de estados e escalas”, são descritas as entidades e os agentes modelados. Na terceira subcategoria, “Resumo do processo”, são apresentados o resumo da execução do modelo e as ações que eles efetuam. Na quarta subcategoria, “Conceitos de *design*”, são apresentados os principais pontos do *design* do modelo. Na quinta subcategoria, “Inicialização”, são descritos quais parâmetros são utilizados na inicialização do sistema. Na sexta subcategoria, “Dados de inicialização”, são definidos os dados e as configurações do sistema na inicialização do modelo. Por fim, na sétima subcategoria, “Submodelos”, são descritas as características de variações do modelo. Na Tabela 4, encontra-se a descrição do modelo desenvolvido neste trabalho de pesquisa, conforme esse protocolo.

Tabela 2 – Diagrama simplificado do modelo, utilizando o protocolo ODD

Inicialização	<p>Entidade: Espaços – Blocos (<i>Patches</i>)</p> <p>Variáveis da Entidade:</p> <ul style="list-style-type: none"> - Clima (se é uma nuvem – cumulonimbus) - Célula Bloqueada (sim ou não) <p>Entidade: Aeronaves (VTOL)</p> <p>Variáveis da Entidade:</p> <ul style="list-style-type: none"> - Coordenadas de destino - Distância percorrida - Tempo percorrido <p>Parâmetros de Entrada do Modelo:</p> <ul style="list-style-type: none"> - Tamanho do Mundo (15NMx15NM ou 30NM x 30NM) - Velocidade Vertical (VTOL1 e VTOL2) - Velocidade Horizontal (VTOL1 e VTOL2) - Número de CBs (CB1 e CB2) - Quantidade de VTOLs de cada tipo (VTOL1 e VTOL2) no intervalo de tempo - Intervalo de tempo em que novas aeronaves são incluídas - Separação estipulada entre aeronaves na vertical - Separação estipulada entre aeronaves na horizontal - Tipo de cenário (perpendicular, paralelo ou geral)
Modelo	<ol style="list-style-type: none"> 1- Aeronave é alocada a uma altitude – nível (dependendo da sua direção), 2- Aeronave bloqueia as células até a altitude designada e sobe verticalmente até essa altitude, 3- Aeronave ajusta a sua proa até a direção que deseja seguir, 4- Aeronave continua seguindo para frente na velocidade escolhida, 5- No caso da existência de nuvens ou de outras aeronaves no mesmo nível, a aeronave tenta manter o espaçamento, reduzindo a velocidade, 6- Em caso de necessidade, a aeronave pode parar no ar, em estado de espera, ou fazer um desvio pela direita ou por baixo, se possível. 7- Quando a aeronave se encontra em cima do destino, tenta bloquear as células abaixo dela e, então, passa a descer verticalmente, desbloqueando as células acima dela.
Análise	<ul style="list-style-type: none"> - Número de aeronaves que conseguiram concluir a viagem até o destino. - Média de tempo e distâncias percorridas pelas aeronaves. - Número de violações de distância e colisões ocorridas durante a simulação.

Tabela 3 – Modelo ODD

O (<i>Overview</i>)	1. Propósito e padrões
	2. Entidades, variáveis de estados e escalas
	3. Resumo do processo
D (<i>Design concepts</i>)	4. Conceitos de <i>design</i>
D (<i>Details</i>)	5. Inicialização
	6. Dados de inicialização
	7. Submodelos

Tabela 4 – Diagrama completo do modelo, utilizando o protocolo ODD.

Categorias	Subcategorias	Descrição
O (<i>Overview</i>)	1. Propósito e padrões	O modelo simula um espaço do UAM entre 1.000 ft e 3.200 ft, para as aeronaves tentarem ir de um ponto a outro dentro desse espaço, com o objetivo de validar regras de capacidade do espaço aéreo. Pode-se alterar parâmetros como taxa de entrada de novas aeronaves e distância mínima entre aeronaves, permitindo detectar gargalos no sistema e analisar como os parâmetros e as regras definidas podem afetar a capacidade do espaço aéreo em geral.
	2. Entidades, variáveis de estados e escalas	<p>Entidade: Espaços – Blocos (<i>Patches</i>)</p> <p>Variáveis da Entidade:</p> <ul style="list-style-type: none"> - Clima (se é uma nuvem – cumulonimbus) - Célula Bloqueada (sim ou não) <p>Entidade: Aeronaves (VTOL)</p> <p>Variáveis da Entidade:</p> <ul style="list-style-type: none"> - Coordenadas de destino - Distância percorrida - Tempo percorrido

Categorias	Subcategorias	Descrição
	3. Resumo do processo	<p>1- Aeronave é alocada a uma altitude – nível (dependendo da sua direção),</p> <p>2- Aeronave bloqueia as células até a altitude designada e sobe verticalmente até essa altitude,</p> <p>3- Aeronave ajusta a sua proa até a direção que deseja seguir,</p> <p>4- Aeronave continua seguindo para frente na velocidade escolhida,</p> <p>5- No caso da existência de nuvens ou de outras aeronaves no mesmo nível, a aeronave tenta manter o espaçamento, reduzindo a velocidade,</p> <p>6- Em caso de necessidade, a aeronave pode parar no ar, em estado de espera, ou fazer um desvio pela direita ou por baixo, se possível.</p> <p>7- Quando a aeronave se encontra em cima do destino, tenta bloquear as células abaixo dela e, então, passa a descer verticalmente, desbloqueando as células acima dela.</p>
<p>D <i>(Design Concepts)</i></p>	4. Conceitos de <i>design</i>	O cenário inicial pode ser de tamanho 15 NM x 15 NM ou 30 NM x 30 NM.
<p>D <i>(Details)</i></p>	5. Inicialização	<p>Parâmetros de Entrada do Modelo:</p> <ul style="list-style-type: none"> - Tamanho do Mundo (15NMx15NM ou 30NM x 30NM) - Velocidade Vertical (VTOL1 e VTOL2) - Velocidade Horizontal (VTOL1 e VTOL2) - Número de CBs (CB1 e CB2) - Quantidade de VTOLs de cada tipo (VTOL1 e VTOL2) no intervalo de tempo - Intervalo de tempo em que novas aeronaves são incluídas - Separação estipulada entre aeronaves na vertical - Separação estipulada entre aeronaves na horizontal - Tipo de cenário (paralelo, geral ou perpendicular)

Categorias	Subcategorias	Descrição
	6. Dados de inicialização	<ul style="list-style-type: none">- Inicialização sem nenhuma aeronave com trajetória em andamento.- Parâmetros de entrada- Dados estocásticos na criação dos CBs durante a inicialização do ambiente.
	7. Submodelos	<ul style="list-style-type: none">- Perpendicular- Paralela- Geral

6. RESULTADOS

Neste capítulo, apresentam-se resultados e análises de cenários executados utilizando-se o modelo multiagente desenvolvido nesse trabalho. Os parâmetros de entrada do modelo são variados nas diversas seções, visando a uma análise de sensibilidade desses parâmetros.

Na primeira seção, são apresentados três cenários visando ilustrar o formato em que os resultados e os cenários estão sendo desenvolvidos, de maneira mais didática, enquanto que, nas demais seções, apresentam-se os resultados de forma consolidada, incluindo cenários mais complexos com formações climáticas de CBs.

Na segunda seção constam 15 cenários, variando-se as velocidades das aeronaves bem como as formas de trajetórias das aeronaves. Na terceira seção, apresentam-se alguns cenários variando o parâmetro de separação entre aeronaves. Na quarta seção, são apresentados dois cenários variando-se a taxa de entrada (quantidade) de aeronaves entrando no modelo por ciclo de simulação. E, por fim, na última seção, é feita uma análise geral dos resultados das simulações, complementando aquelas apresentadas em cada uma dessas seções.

Esses cenários foram escolhidos variando-se parâmetros como velocidade e número de veículos adentrando o ambiente de simulação a cada ciclo. Cada cenário tem como objetivo variar um determinado parâmetro para que se possa, então, ter uma capacidade de comparação entre cenários e poder medir o impacto de cada parâmetro no ambiente.

6.1 PRIMEIRO EXEMPLO

Neste primeiro exemplo são apresentados, inicialmente, três cenários de simulação em um ambiente de 15 NM x 15 NM, nos quais o tempo simulado é sempre de 600 minutos (correspondentes a 10 horas). A finalidade desse exemplo é mostrar as principais características e parâmetros envolvidos no modelo, permitindo a comparação entre estes três cenários. Todos esses cenários utilizam a navegação do cenário do tipo “geral”, conforme descrito no capítulo. Posteriormente, outros cenários de simulação serão realizados, variando-se outros parâmetros do modelo e utilizando a mesma metodologia aqui apresentada.

Na Tabela 5 encontram-se alguns parâmetros fixados para os três cenários simulados neste primeiro exemplo, como a duração de um ciclo (intervalo de geração de novas aeronaves) igual a 3 minutos, e a quantidade máxima de agentes VTOL dos tipos 1 e 2 que podem ser inseridos o ambiente de simulação a cada novo ciclo.

Tabela 5 – Configuração de parâmetros do primeiro exemplo: Número de agentes e ciclo

Parâmetro	Configuração
Número Máximo de VTOL 1	3
Número Máximo de VTOL 2	1
Intervalo de Geração – ciclo (min)	3

Fonte: Autor (2022)

Na Tabela 6, constam as definições de separação mínima (Horizontal e Vertical) usadas durante o exemplo.

Tabela 6 – Configuração de parâmetros do primeiro exemplo: Separação

Parâmetro	Configuração
Separação Horizontal definida (NM)	1
Separação Vertical definida (ft)	200

Fonte: Autor (2022)

Nas Tabelas 7, 8 e 9 apresentam-se, respectivamente, para cada um dos três cenários, as definições dos parâmetros de velocidade (vertical e horizontal). No cenário 1, o VTOL 1 é mais lento do que o VTOL 2 (sendo que o VTOL 1 existe em maior quantidade no ambiente); no cenário 2, a situação é inversa, ou seja, o VTOL 1 é mais rápido do que o VTOL 2; e, por fim, no cenário 3, o VTOL 1 é muito mais rápido (5 vezes mais rápido) do que o VTOL 2. Cada um dos 3 cenários foi simulado 100 vezes, gerando-se, portanto, 100 execuções do modelo para cada cenário.

Tabela 7 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 1

Parâmetro	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	200	400
Velocidade Horizontal (KT)	30	60

Fonte: Autor (2022).

Tabela 8 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 2

Parâmetro	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	400	200
Velocidade Horizontal (KT)	60	30

Fonte: Autor (2022).

Tabela 9 – Configuração de parâmetros do primeiro exemplo: Velocidades no cenário 3

Parâmetro	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	1.000	200
Velocidade Horizontal (KT)	150	30

Fonte: Autor (2022).

As Tabelas 10 e 11 apresentam os resultados das primeiras 10 execuções de simulação feitas com o cenário 1 (de um total de 100), cada uma com duração de 600 minutos. Cada linha, numerada de 01 a 10, exemplifica uma dessas execuções. Na Tabela 10 (parte 1) constam 4 variáveis de saída do modelo: viagens completadas, número de violações de separação mínima, número de violações de separação estipulada e número de veículos atrasados (em *delay*).

Tabela 10 – Execuções da simulação: Cenário 1 (parte 1)

Execução	Viagens Completadas	Casos Abaixo da Separação Mínima	Casos Abaixo da Separação Estipulada	Número de Veículos Atrasados
01	310	0	7	0
02	308	0	0	3
03	301	0	9	0
04	289	0	6	1
05	296	0	2	10
06	275	2	0	1
07	254	0	367	8
08	293	0	357	8
09	253	0	4	12
10	273	0	5	0

Fonte: Autor (2022).

Na Tabela 11 (parte 2) constam outras 5 variáveis de saída do modelo, considerando as viagens completadas realizadas pelas aeronaves durante o período de simulação: distância média percorrida na vertical (**em pés - ft**), distância média percorrida na horizontal (**em milhas náuticas – NM**), tempo médio das viagens (**em**

minutos), tempo médio do percurso vertical das viagens (**em minutos**) e tempo médio do percurso horizontal das viagens (**em minutos**). Essas unidades valem para todas as demais tabelas similares deste capítulo.

Tabela 11 – Execuções da simulação: Cenário 1 (parte 2)

Execução	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
01	2610,32	3,86	13,15	11,45	1,70
02	2449,35	4,74	12,56	10,74	1,83
03	2706,98	4,77	13,44	11,84	1,60
04	2408,30	3,35	11,99	10,54	1,45
05	2613,51	6,02	13,57	11,70	1,87
06	2510,55	4,45	12,83	11,14	1,69
07	2511,81	4,10	12,57	11,06	1,51
08	2565,19	3,94	12,79	11,26	1,53
09	2657,71	5,26	13,79	11,86	1,93
10	2457,14	4,12	12,10	10,58	1,52

Fonte: Autor (2022)

As Tabelas 12 e 13 apresentam dados consolidados gerados a partir das informações de 100 execuções do modelo, para o cenário 1, sendo que as 10 primeiras dessas simulações foram apresentadas nas Tabelas 10 e 11. Para cada parâmetro de saída foram gerados dados estatísticos, referentes à média, à moda, ao desvio padrão, ao valor máximo, ao valor mínimo e aos 1º e 3º quartis dos dados. Assim, é possível analisar melhor os resultados das simulações, permitindo comparações entre os cenários.

Tabela 12 – Resumo dos resultados do Cenário 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	286,30	0,38	87,86	3,51
Moda	284	0	5	3
Desvio- Padrão	15,42	0,79	220,90	3,03
Máximo	318	2	1076	17
Mínimo	247	0	0	0
1º Quartil	275,75	0	3	1
3º Quartil	296	0	12	5

Fonte: Autor (2022)

Tabela 13 – Resumo dos resultados do Cenário 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2574,75	4,18	13,02	11,40	1,66
Moda	2539,50	4,41	13	11,45	1,69
Desvio- Padrão	85,32	0,78	0,51	0,45	0,19
Máximo	2804,07	6,10	14,80	13,05	2,13
Mínimo	2375,08	2,77	11,99	10,47	1,21
1º Quartil	2517,70	3,64	12,70	11,13	1,52
3º Quartil	2633,04	4,69	13,38	11,67	1,77

Fonte: Autor (2022)

As Tabelas 14 e 15 apresentam dados consolidados gerados a partir das informações de 100 execuções do modelo, para o cenário 2. Para cada parâmetro de saída foram gerados dados estatísticos, referentes à média, à moda, ao desvio padrão, ao valor máximo, ao valor mínimo e aos 1º e 3º quartis dos dados. Assim, é possível analisar melhor os resultados das simulações, permitindo comparações entre os cenários.

Tabela 14 – Resumo dos resultados do Cenário 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	290,89	0,42	87,77	2,77
Moda	283	0	0	0
Desvio-Padrão	14,12	0,91	253,36	2,42
Máximo	323	4	1172	10
Mínimo	252	0	0	0
1º Quartil	282,75	0	1	1
3º Quartil	302	0	8,25	4

Fonte: Autor (2022)

Tabela 15 – Resumo dos resultados do Cenário 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2574,09	4,54	9,95	8,45	1,50
Moda	-	3,74	10,02	8,39	1,41
Desvio-Padrão	78,48	1,05	0,42	0,35	0,19
Máximo	2771,83	9,02	12,09	10,19	2,08
Mínimo	2375,23	2,68	9,10	7,79	0,98
1º Quartil	2523,22	3,77	9,67	8,22	1,35
3º Quartil	2631,93	5,11	10,18	8,66	1,63

Fonte: Autor (2022)

As Tabelas 16 e 17 apresentam dados consolidados gerados a partir das informações de 100 execuções do modelo, para o cenário 3, considerando os mesmos parâmetros de saída estatísticos considerados nos cenários 1 e 2.

Tabela 16 – Resumo dos resultados do Cenário 3 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	294,24	0,38	13,63	1,45
Moda	292	0	2	0
Desvio-Padrão	13,94	1,05	75,71	2,07
Máximo	324	6	732	7
Mínimo	264	0	0	0
1º Quartil	285,75	0	1	0
3º Quartil	304,25	0	5,25	3

Fonte: Autor (2022)

Tabela 17 – Resumo dos resultados do Cenário 3 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2596,40	7,31	7,52	5,84	1,69
Moda	-	6,67	7,63	5,85	1,54
Desvio- Padrão	79,90	2,15	0,46	0,30	0,37
Máximo	2789,74	16,78	8,95	6,70	3,35
Mínimo	2377,14	4,80	6,64	5,23	1,19
1º Quartil	2552,76	6,05	7,18	5,60	1,45
3º Quartil	2639,39	7,84	7,77	6,05	1,79

Fonte: Autor (2022)

Uma comparação inicial dos parâmetros de saída do modelo, considerando os 3 cenários apresentados, pode ser realizada por meio de gráficos com o uso do *box-plot*, que permitem visualizar as variações da distribuição estatística desses parâmetros.

A Figura 53 compara o número de viagens completadas, considerando os cenários 1, 2 e 3. Pode-se observar que, apesar de o parâmetro de entrada Velocidade dos VTOLs ter variado bastante entre os cenários, o número de viagens completadas não aumentou na mesma proporção.

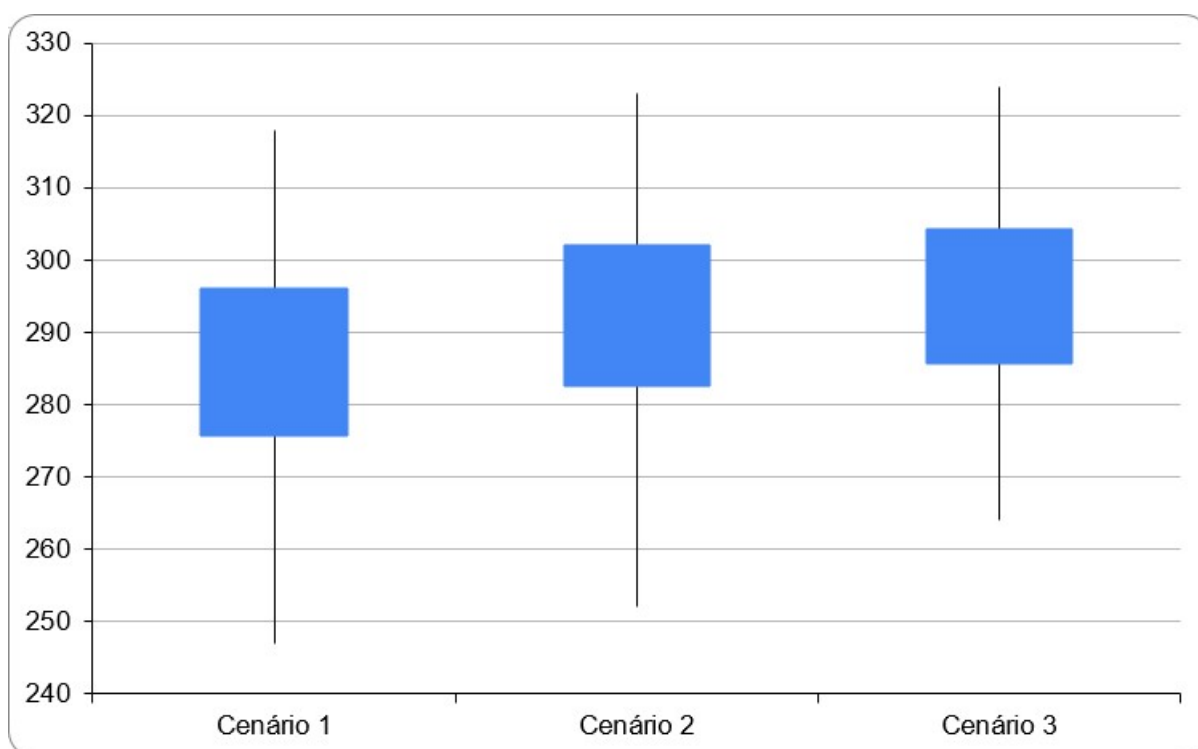


Figura 53 – Viagens Completadas por Cenário (Número de Veículos Concluintes)

Fonte: Autor (2022)

A Figura 54 apresenta a comparação do tempo médio das viagens realizadas em cada um dos 3 cenários simulados. Neste caso, considerando as variações do parâmetro de entrada Velocidade dos VTOLs nos cenários simulados, houve uma redução significativa dos tempos médios das viagens completadas.

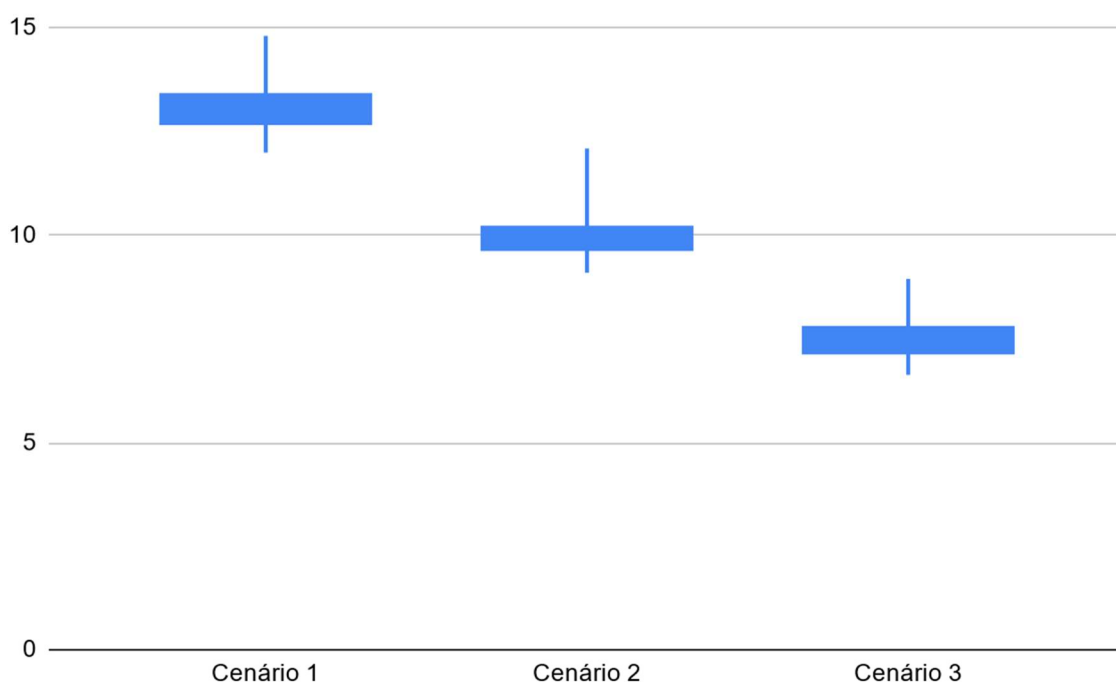


Figura 54 – Tempo Médio das Viagens por Cenário (tempo em minutos)

Fonte: Autor (2022)

Na Figura 55 apresenta-se a comparação da variação do número de violações de separação mínima ocorridas em função de cada um dos três cenários simulados. Percebe-se que, com o aumento de velocidade dos VTOLs, e a consequente menor probabilidade de existirem mais aeronaves voando simultaneamente no espaço simulado, essas ocorrências de violação de separação tendem a diminuir. Vale ressaltar que os gráficos se mostram distorcidos pela existência de pelo menos uma situação de *outlier* que gerou um caso em que o modelo entrou em uma situação de crise, quando muitas aeronaves ficam localizadas em um mesmo setor do ambiente de simulação.

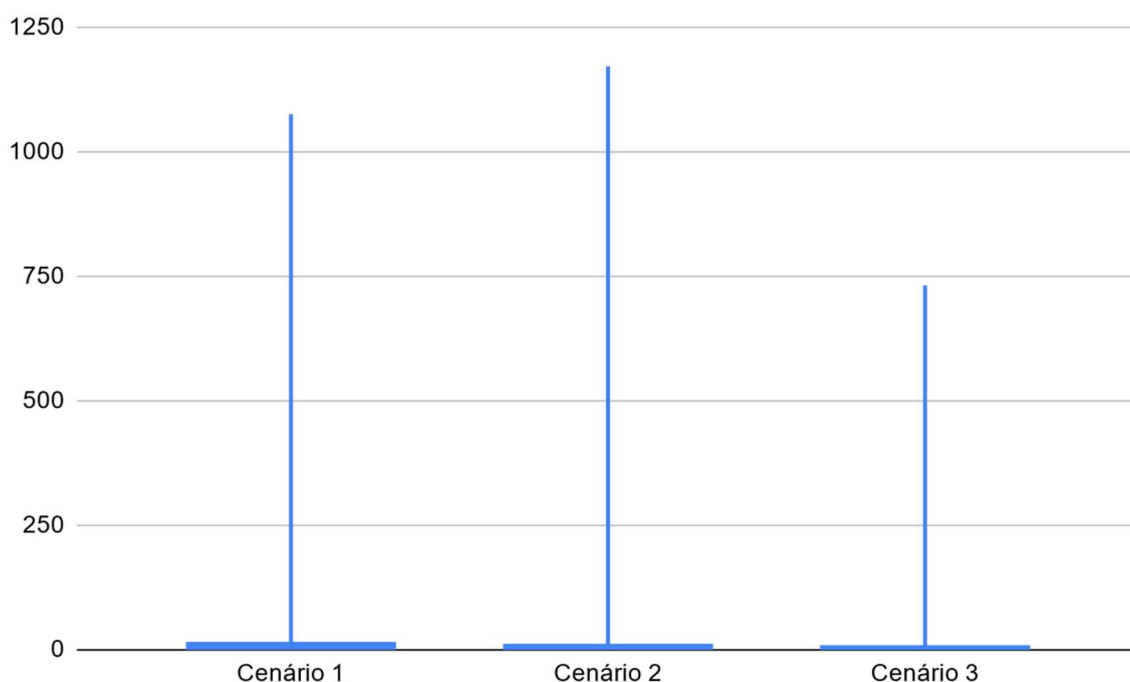


Figura 55 – Número de Situações de Ocorrência de Violação da Separação Mínima por Cenário
Fonte: Autor (2022)

Por fim, para finalizar a análise comparativa das simulações dos cenários 1, 2 e 3, pode-se avaliar que no cenário 1 existem aeronaves maiores e, portanto, mais lentas voando no espaço aéreo simulado. Isso tende a gerar maior tempo de espera (“*delay*”) de algumas aeronaves e, portanto, um maior tempo médio de conclusão de viagens. Por outro lado, com o aumento da velocidade dos VTOLs, como apresentado no cenário 3, ocorre uma redução no tempo médio das viagens realizadas, o que contribui para a diminuição do número de violações de separação estipuladas, já que o espaço aéreo simulado fica menos congestionado, com menos aeronaves voando simultaneamente. Apesar disso, quanto às separações mínimas, houve um pior caso mais grave no cenário 3 do que o pior caso do cenário 1, o que mostra que, apesar de se ter desafogado o tráfego de aeronaves no cenário 3, e de se ter gerado um menor número de violações estipuladas, acabou-se por gerar uma probabilidade maior da existência de um acidente nesse cenário.

6.2 DIFERENTES TIPOS DE CENÁRIOS

Nesta seção são realizadas simulações com 15 diferentes tipos de cenários, considerando em todos eles: ambiente de simulação de 30 NM x 30 NM; ciclo de 3 minutos, até 3 VTOLs do tipo 1 ingressando a cada ciclo; até 1 VTOL do tipo 2 ingressando a cada ciclo. Além disso, em todos esses 15 cenários os parâmetros de separação adotados são os mesmos utilizados nas simulações dos cenários 1, 2 e 3, ou seja: separação horizontal definida de 1 NM e separação vertical definida de 200 ft.

Nos cinco primeiros cenários (4 a 8) utiliza-se o tipo de trajetória “**paralela**” a ser adotada pelas aeronaves no ambiente; nos 5 cenários seguintes (9 a 13) as aeronaves adotam o tipo de trajetória “**perpendicular**”; e, por fim, nos 5 últimos cenários (14 a 18) as aeronaves adotam o tipo de trajetória “**geral**”.

Os cinco cenários em cada um desses três grupos de cenários são rotulados da seguinte maneira: Caso 1, Caso 2, Caso 3, Caso 2 com CB-1 e Caso 2 com CB-2, o que será explicado a seguir. Os casos 1, 2 e 3 utilizam parâmetros de velocidade (vertical em ft/min e horizontal em KT) dos dois tipos de VTOLs conforme o apresentado nas Tabelas 18, 19 e 20.

Tabela 18 – Velocidades dos VTOLs – Caso 1

	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	200	200
Velocidade Horizontal (KT)	30	30

Fonte: Autor (2022)

Tabela 19 – Velocidades dos VTOLs – Caso 2

	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	400	200
Velocidade Horizontal (KT)	60	30

Fonte: Autor (2022)

Tabela 20 – Velocidades dos VTOLs – Caso 3

	VTOL 1	VTOL 2
Velocidade Vertical (ft/min)	800	200
Velocidade Horizontal (KT)	90	30

Fonte: Autor (2022)

O Caso 2 com CB-1 trata-se de um cenário equivalente ao cenário do Caso 2, porém existe a introdução de formações climáticas de Cumulonimbus do tipo 1. Já o Caso 2 com CB-2 trata-se de um cenário equivalente ao cenário do Caso 2, porém existe a introdução de formações climáticas de Cumulonimbus do tipo 2. Os cenários que envolvem formações climáticas de Cumulonimbus consideram exatamente a existência de 25 formações dessa natureza inseridas no espaço de simulação de forma estocástica a cada execução de uma simulação.

Nas tabelas de 21 a 30 a seguir apresentam-se os resultados consolidados das simulações que utilizam trajetória “**paralela**” para as aeronaves. Em particular, tem-se o seguinte:

- As tabelas 21 e 22: apresentam o resultado consolidado das simulações executadas para o cenário 4 (Caso 1);
- As tabelas 23 e 24: apresentam o resultado consolidado das simulações executadas para o cenário 5 (Caso 2);
- As tabelas 25 e 26: apresentam o resultado consolidado das simulações executadas para o cenário 6 (Caso 3);
- As tabelas 27 e 28: apresentam o resultado consolidado das simulações executadas para o cenário 7 (Caso 2 com CB-1);
- As tabelas 29 e 30: apresentam o resultado consolidado das simulações executadas para o cenário 8 (Caso 2 com CB-2).

Tabela 21 – Trajetória Paralela: Resultados do Cenário 4 – Caso 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	280,68	0	62,28	0,94
Moda	280	0	6	0
Desvio- Padrão	13,83	0	148,24	1,40
Máximo	314	0	655	6
Mínimo	246	0	0	0
1º Quartil	272,75	0	3	0
3º Quartil	288,25	0	9	2

Fonte: Autor (2022)

Tabela 22 – Trajetória Paralela: Resultados do Cenário 4 – Caso 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2582,18	9,94	32,82	12,91	19,91
Moda	2593,77	10,25	33,19	12,57	19,58
Desvio-Padrão	88,35	0,48	1,01	0,44	0,96
Máximo	2801,43	11,13	35,41	14,01	22,37
Mínimo	2357,65	8,56	29,94	11,79	17,12
1º Quartil	2514,02	9,65	32,17	12,57	19,32
3º Quartil	2643,30	10,25	33,42	13,21	20,52

Fonte: Autor (2022)

Tabela 23 – Trajetória Paralela: Resultados do Cenário 5 – Caso 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	288,40	0,04	32,77	0,60
Moda	291	0	0	0
Desvio-Padrão	14,65	0,4	115,34	1,16
Máximo	321	4	886	5
Mínimo	240	0	0	0
1º Quartil	279,75	0	1	0
3º Quartil	298	0	3	0,25

Fonte: Autor (2022)

Tabela 24 – Trajetória Paralela: Resultados do Cenário 5 – Caso 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2603,83	10,06	21,16	8,46	12,70
Moda	2652,26	10,19	21,14	8,54	12,88
Desvio- Padrão	76,43	0,46	0,75	0,28	0,62
Máximo	2808,05	11,08	23	9,08	14,21
Mínimo	2424,41	8,95	19,53	7,70	11,26
1º Quartil	2554,37	9,69	20,59	8,28	12,24
3º Quartil	2652,62	10,37	21,71	8,66	13,14

Fonte: Autor (2022)

Tabela 25 – Trajetória Paralela: Resultados do Cenário 6 – Caso 3 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	290,06	0,08	28,91	0,3
Moda	292	0	1	0
Desvio- Padrão	14,02	0,39	100,50	0,76
Máximo	326	2	529	4
Mínimo	253	0	0	0
1º Quartil	281	0	0	0
3º Quartil	299,25	0	2	0

Fonte: Autor. (2022)

Tabela 26 – Trajetória Paralela: Resultados do Cenário 6 – Caso 3 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2588,81	10,15	16,36	6,13	10,19
Moda	2595,40	10,09	16,33	6,15	10,18
Desvio-Padrão	77,012	0,44	0,66	0,25	0,53
Máximo	2787,60	11,08	18	6,97	11,56
Mínimo	2398,57	8,95	14,89	5,72	9,14
1º Quartil	2545,55	9,73	15,85	5,99	9,79
3º Quartil	2650,84	10,39	16,74	6,31	10,54

Fonte: Autor (2022)

Tabela 27 – Trajetória Paralela: Resultados do Cenário 7 – Caso 2 com CB-1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	276,14	0,14	34,03	12,72
Moda	283	0	1	11
Desvio-Padrão	13,58	0,57	99,25	3,91
Máximo	310	4	550	27
Mínimo	242	0	0	6
1º Quartil	267	0	1	10
3º Quartil	285	0	5	15

Fonte: Autor (2022)

Tabela 28 – Trajetória Paralela: Resultados do Cenário 7 – Caso 2 com CB-1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2596,02	9,71	20,75	8,43	12,32
Moda	2484,21	9,33	20,52	8,50	11,95
Desvio-Padrão	74,36	0,38	0,64	0,25	0,53
Máximo	2780,21	10,52	22,51	9,08	13,85
Mínimo	2441,32	8,84	19,21	7,82	11,23
1º Quartil	2551,75	9,44	20,30	8,25	11,95
3º Quartil	2648,68	10,01	21,25	8,61	12,66

Fonte: Autor. (2022)

Tabela 29 – Trajetória Paralela: Resultados do Cenário 8 – Caso 2 com CB-2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	272,17	0,10	44,05	12,96
Moda	278	0	0	11
Desvio-Padrão	12,67	0,44	115,47	3,74
Máximo	301	2	587	23
Mínimo	241	0	0	5
1º Quartil	264	0	1	11
3º Quartil	282	0	6	15

Fonte: Autor (2022)

Tabela 30 – Trajetória Paralela: Resultados do Cenário 8 – Caso 2 com CB-2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2586,71	9,87	20,95	8,45	12,50
Moda	-	9,74	20,52	8,60	12,02
Desvio- Padrão	82,33	0,44	0,79	0,31	0,65
Máximo	2785,88	10,88	23,24	9,11	14,33
Mínimo	2362,03	8,82	19,19	7,76	11,23
1º Quartil	2535,50	9,57	20,47	8,23	12,04
3º Quartil	2637,66	10,155	21,40	8,70	12,823

Fonte: Autor (2022)

Pode-se observar que, nos cenários 7 e 8, com a adição de CBs, tem-se uma redução no número de viagens concluídas no período simulado, e um aumento do número de viagens com atraso (para as execuções com os piores casos). Com introdução de formações climáticas nas simulações com trajetórias “**paralelas**” das aeronaves não houve a geração de muitos conflitos entre aeronaves, trazendo um impacto pequeno nesse quesito. Isso se deve ao fato de que, como as trajetórias das aeronaves são predominantemente paralelas, os conflitos não são potencializados mesmo com a introdução das formações climáticas.

Também se pode perceber que, nos cenários 4, 5 e 6, houve com um aumento crescente da velocidade média desenvolvidas pelas aeronaves, e a consequente redução na média do tempo de viagem das aeronaves, à medida que a velocidade máxima dos VTOLs cresce do cenário 4 para o cenário 5, e também do cenário 5 para o cenário 6. Em contrapartida, à medida que as velocidades máximas desenvolvidas pelas aeronaves crescem, aumenta o número de violações de separação mínima. Deve-se observar, por outro lado, que o aumento do número de conclusões de viagens não é significativo, pois essa variável não é tão dependente das velocidades das aeronaves, mas sim do número de aeronaves que entram no ambiente de simulação por unidade de tempo.

Nas tabelas de 31 a 40 a seguir, apresentam-se os resultados consolidados das simulações que utilizam trajetória “**perpendicular**” para as aeronaves. Em particular, tem-se o seguinte:

- As tabelas 31 e 32: apresentam o resultado consolidado das simulações executadas para o cenário 9 (Caso 1);
- As tabelas 33 e 34: apresentam o resultado consolidado das simulações executadas para o cenário 10 (Caso 2);
- As tabelas 35 e 36: apresentam o resultado consolidado das simulações executadas para o cenário 11 (Caso 3);
- As tabelas 37 e 38: apresentam o resultado consolidado das simulações executadas para o cenário 12 (Caso 2 com CB-1);
- As tabelas 39 e 40: apresentam o resultado consolidado das simulações executadas para o cenário 13 (Caso 2 com CB-2).

Tabela 31 – Trajetória Perpendicular: Resultados do Cenário 9 – Caso 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	279,07	0,04	74,13	1,36
Moda	276	0	5	0
Desvio- Padrão	13,88	0,28	182,32	1,98
Máximo	314	2	935	10
Mínimo	244	0	0	0
1º Quartil	270	0	4	0
3º Quartil	289	0	10,25	2

Fonte: Autor (2022)

Tabela 32 – Trajetória Perpendicular: Resultados do Cenário 9 – Caso 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2590,94	9,94	32,39	12,96	19,44
Moda	2531,93	10,03	32,20	13,19	19,63
Desvio-Padrão	83,05	0,39	0,85	0,43	0,79
Máximo	2836,09	10,89	34,55	14,18	21,37
Mínimo	2401,45	8,75	29,62	12,02	17,02
1º Quartil	2531,02	9,65	31,82	12,66	18,84
3º Quartil	2642,33	10,25	32,93	13,21	20,03

Fonte: Autor (2022)

Tabela 33 – Trajetória Perpendicular: Resultados do Cenário 10 – Caso 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	288,55	0,08	40,31	0,71
Moda	297	0	0	0
Desvio-Padrão	14,78	0,39	142,86	1,46
Máximo	323	2	107	8
Mínimo	250	0	0	0
1º Quartil	280	0	1	0
3º Quartil	298	0	5	1

Fonte: Autor (2022)

Tabela 34 – Trajetória Perpendicular: Resultados do Cenário 10 – Caso 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2587,35	9,95	20,52	8,42	12,10
Moda	-	9,88	20,98	8,45	12,01
Desvio-Padrão	86,43	0,43	0,68	0,32	0,57
Máximo	2815,33	10,97	22,70	9,25	13,84
Mínimo	2366,22	8,93	18,83	7,63	10,71
1º Quartil	2530,82	9,65	20,09	8,22	11,74
3º Quartil	2641,47	10,25	20,98	8,65	12,44

Fonte: Autor (2022)

Tabela 35 – Trajetória Perpendicular: Resultados do Cenário 11 – Caso 3 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	290,91	0	7,74	0,29
Moda	298	0	0	0
Desvio-Padrão	16,54	0	40,61	0,86
Máximo	336	0	290	4
Mínimo	255	0	0	0
1º Quartil	279	0	0	0
3º Quartil	302	0	2	0

Fonte: Autor (2022)

Tabela 36 – Trajetória Perpendicular: Resultados do Cenário 11 – Caso 3 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2599,71	10,04	15,89	6,21	9,69
Moda	2578,87	10,04	15,49	6,13	9,50
Desvio-Padrão	78,05	0,47	0,78	0,28	0,64
Máximo	2806,69	11,11	18,28	6,89	11,72
Mínimo	2430,87	8,92	13,41	5,52	7,89
1º Quartil	2550,85	9,74	15,36	6,03	9,34
3º Quartil	2661,51	10,38	16,42	6,36	10,08

Fonte: Autor (2022)

Tabela 37 – Trajetória Perpendicular: Resultados do Cenário 12 – Caso 2 com CB-1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	275,37	0,14	21,66	11,48
Moda	267	0	2	9
Desvio-Padrão	13,81	0,59	76,55	3,76
Máximo	312	4	582	21
Mínimo	247	0	0	4
1º Quartil	267	0	1	9
3º Quartil	285	0	5	14

Fonte: Autor (2022)

Tabela 38 – Trajetória Perpendicular: Resultados do Cenário 12 – Caso 2 com CB-1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2596,27	9,77	20,27	8,45	11,83
Moda	2401,43	9,38	19,59	8,13	11,94
Desvio-Padrão	84,33	0,47	0,77	0,31	0,67
Máximo	2827,54	10,74	22,60	9,35	13,51
Mínimo	2401,43	8,65	18,71	7,64	10,37
1º Quartil	2539,17	9,41	19,83	8,24	11,41
3º Quartil	2653,51	10,09	20,71	8,61	12,22

Fonte: Autor (2022)

Tabela 39 – Trajetória Perpendicular: Resultados do Cenário 13 – Caso 2 com CB-2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	272,88	0,14	37,25	13,39
Moda	275	0	1	10
Desvio-Padrão	13,33	0,59	103,92	4,06
Máximo	306	4	643	26
Mínimo	242	0	0	4
1º Quartil	264	0	1	10
3º Quartil	282	0	6	16

Fonte: Autor (2022)

Tabela 40 – Trajetória Perpendicular: Resultados do Cenário 13 – Caso 2 com CB-2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2593,86	9,82	20,36	8,44	11,92
Moda	2516,13	10,12	20,26	8,36	11,39
Desvio- Padrão	83,52	0,40	0,72	0,31	0,57
Máximo	2773,82	11,04	22,74	9,14	13,81
Mínimo	2330,57	8,88	18,94	7,71	10,74
1º Quartil	2548,09	9,53	19,95	8,26	11,49
3º Quartil	2644,31	10,09	20,84	8,63	12,27

Fonte: Autor (2022)

Comparando-se os cenários de 9 a 11 (casos de 1 a 3 com trajetórias “**perpendiculares**”) com os cenários de 4 a 6 (casos de 1 a 3 com trajetórias “**paralelas**”), pode-se observar o seguinte: a) nas simulações com trajetórias “**perpendiculares**”, ocorre um aumento no número de conflitos entre aeronaves, já que, nesses tipos de cenários, existem cruzamentos entre rotas de aeronaves em um mesmo nível de voo; b) nas simulações com trajetórias “**perpendiculares**”, ocorre um aumento na média do tempo de viagem das aeronaves e uma redução do número de conclusões de viagens, pois torna-se mais comum aeronaves terem que reduzir suas velocidades ou ficar em estado de espera (“*delay*”) para aguardar outra aeronave terminar de cruzar o seu caminho.

Por outro lado, ocorre uma relativa melhora nos cenários 12 e 13 (casos 2 com CBs, com trajetórias “**perpendiculares**”), quando comparados aos cenários 7 e 8 (casos 2 com CBs, com trajetórias “**paralelas**”), apesar do aumento da complexidade do espaço aéreo em função da introdução de formações climáticas, no tocante aos tempos médios de conclusões de viagens. Como existem dois eixos perpendiculares para as aeronaves trafegarem em um mesmo nível de voo, passam a existir mais alternativas para elas concluírem suas viagens.

Nas tabelas de 41 a 50 a seguir apresentam-se os resultados consolidados das simulações que utilizam trajetória “**geral**” para as aeronaves. Em particular, tem-se o seguinte:

- As tabelas 41 e 42: apresentam o resultado consolidado das simulações executadas para o cenário 14 (Caso 1);
- As tabelas 43 e 44: apresentam o resultado consolidado das simulações executadas para o cenário 15 (Caso 2);
- As tabelas 45 e 46: apresentam o resultado consolidado das simulações executadas para o cenário 16 (Caso 3);
- As tabelas 47 e 48: apresentam o resultado consolidado das simulações executadas para o cenário 17 (Caso 2 com CB-1);
- As tabelas 49 e 50: apresentam o resultado consolidado das simulações executadas para o cenário 18 (Caso 2 com CB-2).

Tabela 41 – Trajetória Geral: Resultados do Cenário 14 – Caso 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	289,81	0,24	49,62	0,81
Moda	294	0	2	0
Desvio- Padrão	14,17	0,65	235,18	1,52
Máximo	345	2	1971	9
Mínimo	261	0	0	0
1º Quartil	280	0	3,75	0
3º Quartil	298,25	0	10	1

Fonte: Autor (2022)

Tabela 42 – Trajetória Geral: Resultados do Cenário 14 – Caso 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2566,28	14,89	15,72	12,85	2,87
Moda	-	10,31	15,34	12,66	2,76
Desvio-Padrão	82,39	4,22	0,57	0,42	0,38
Máximo	2744,59	32,2	17,13	13,73	4,5
Mínimo	2391,64	8,71	14,43	11,96	2,22
1º Quartil	2514	11,82	15,30	12,59	2,60
3º Quartil	2618,09	16,65	16,08	13,12	3,09

Fonte: Autor (2022)

Tabela 43 – Trajetória Geral: Resultados do Cenário 15 – Caso 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	294,49	0,08	28,75	0,67
Moda	299	0	0	0
Desvio-Padrão	15,59	0,39	151,52	1,26
Máximo	328	2	1179	5
Mínimo	257	0	0	0
1º Quartil	284,75	0	0	0
3º Quartil	306	0	4	1

Fonte: Autor (2022)

Tabela 44 – Trajetória Geral: Resultados do Cenário 15 – Caso 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2588,85	16,14	10,96	8,46	2,50
Moda	-	13,38	11,01	8,52	2,47
Desvio-Padrão	75,44	4,36	0,45	0,29	0,33
Máximo	2825,42	37,26	13,14	9,36	3,78
Mínimo	2384,81	9,35	9,85	7,71	1,88
1º Quartil	2543,02	13,39	10,71	8,29	2,28
3º Quartil	2632,07	18,16	11,21	8,62	2,68

Fonte: Autor (2022)

Tabela 45 – Trajetória Geral: Resultados do Cenário 16 – Caso 3 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	293,61	0,10	3,95	0,37
Moda	293	0	0	0
Desvio-Padrão	13,347	0,44	16,71	1,02
Máximo	334	2	167	5
Mínimo	255	0	0	0
1º Quartil	283	0	0	0
3º Quartil	303,25	0	4	0

Fonte: Autor (2022)

Tabela 46 – Trajetória Geral: Resultados do Cenário 16 – Caso 3 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2586,12	18,14	8,77	6,18	2,59
Moda	-	15,89	8,68	6,07	2,72
Desvio-Padrão	72,33	3,61	0,43	0,27	0,30
Máximo	2749,09	29,57	10	6,98	3,51
Mínimo	2374,92	9,63	7,92	5,52	1,87
1º Quartil	2540,18	15,83	8,48	6,01	2,38
3º Quartil	2639,16	20,68	9,07	6,36	2,79

Fonte: Autor (2022)

Tabela 47 – Trajetória Geral: Resultados do Cenário 17 – Caso 2 com CB-1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	292,14	0,16	50,78	2,26
Moda	291	0	0	1
Desvio-Padrão	15,46	0,55	194,04	1,76
Máximo	326	2	1055	6
Mínimo	255	0	0	0
1º Quartil	283	0	0	1
3º Quartil	301	0	4	4

Fonte: Autor (2022)

Tabela 48 – Trajetória Geral: Resultados do Cenário 17 – Caso 2 com CB-1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2597,43	15,91	10,97	8,45	2,52
Moda	-	16,23	10,50	8,45	2,63
Desvio-Padrão	84,74	4,54	0,50	0,31	0,33
Máximo	2850,86	40,99	12,27	9,30	3,71
Mínimo	2407,02	9,62	9,84	7,80	1,65
1º Quartil	2550,87	13,02	10,59	8,23	2,29
3º Quartil	2646,49	17,88	11,30	8,64	2,69

Fonte: Autor (2022)

Tabela 49 – Trajetória Geral: Resultados do Cenário 18 – Caso 2 com CB-2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	287,46	0,10	30,79	5,18
Moda	290	0	0	5
Desvio-Padrão	15,50	0,44	119,28	2,89
Máximo	322	2	758	13
Mínimo	242	0	0	0
1º Quartil	277,75	0	0	3
3º Quartil	298,25	0	5	7

Fonte: Autor (2022)

Tabela 50 – Trajetória Geral: Resultados do Cenário 18 – Caso 2 com CB-2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2588,27	15,59	10,93	8,47	2,47
Moda	-	13,51	10,78	8,63	2,25
Desvio- Padrão	83,31	6,55	0,51	0,32	0,39
Máximo	2838,10	54,76	12,62	9,66	4,26
Mínimo	2356,23	9,28	10,03	7,69	1,69
1º Quartil	2535,01	12,46	10,66	8,26	2,25
3º Quartil	2638,97	16,41	11,20	8,63	2,57

Fonte: Autor (2022)

Comparando-se os cenários de 14 a 16 (casos de 1 a 3 com trajetórias “**gerais**”) com os cenários de 4 a 6 (casos de 1 a 3 com trajetórias “**paralelas**”), pode-se observar o seguinte: apesar do aumento da complexidade das trajetórias das aeronaves, houve um aumento significativo da capacidade do espaço aéreo, pois nesses novos cenários ocorreu um aumento no número de viagens completadas, apesar do aumento potencial do número de conflitos.

Comparando-se esses mesmos cenários de 14 a 16 (casos de 1 a 3 com trajetórias “**gerais**”) com os cenários de 9 a 11 (casos de 1 a 3 com trajetórias “**perpendiculares**”), o aumento da capacidade do espaço aéreo não é tão significativo quanto a melhoria em relação aos cenários com trajetórias “**paralelas**”.

Ainda, este conjunto de cenários com CBs (cenários 17 e 18), quando comparado com os cenários 12 e 13, foi menos impactado com a presença de CBs, pela mesma razão apresentada anteriormente: a existência de múltiplos eixos de navegação em um mesmo nível de voo.

6.3 RESULTADOS GERAL – SEPARAÇÃO

Nesta seção são executadas simulações para mais dois cenários (19 e 20), que são comparados com os resultados obtidos no cenário 14, fixado como base, que utiliza trajetórias das aeronaves do tipo “**geral**”, assim como estes dois novos cenários. Entre esses cenários variam-se as separações horizontal (cenário 19) e vertical (cenário 20) recomendadas, conforme apresentado na Tabela 51.

Tabela 51 – Lista de Cenários de Separação

Cenário	Separação Recomendada Horizontal (NM)	Separação Recomendada Vertical (ft)
14	1	200
19	2	200
20	1	400

Fonte: Autor (2022)

Nas tabelas 52 e 53 apresentam-se os resultados da execução simulação do cenário 19, enquanto que nas tabelas 54 e 55 constam os resultados da execução da simulação referente ao cenário 20.

Tabela 52 – Separação: Resultados do Cenário 19 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	289,49	0,10	38,13	0,85
Moda	287	0	9	0
Desvio-Padrão	14,19	0,44	127,31	1,57
Máximo	324	2	897	8
Mínimo	256	0	0	0
1º Quartil	279,75	0	4,75	0
3º Quartil	298	0	13	2

Fonte: Autor (2022)

Tabela 53 – Separação: Resultados do Cenário 19 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2593,44	15,60	15,91	12,99	2,92
Moda	-	16,04	15,86	12,78	2,82
Desvio- Padrão	73,17	4,63	0,55	0,39	0,37
Máximo	2795,70	32,20	17,35	14,02	3,91
Mínimo	2456,16	7,52	14,77	12,28	1,97
1º Quartil	2533,99	12,86	15,49	12,67	2,69
3º Quartil	2646,25	17,59	16,25	13,27	3,11

Fonte: Autor (2022)

Tabela 54 – Separação: Resultados do Cenário 20 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	294,85	0,18	36,36	0,47
Moda	297	0	6	0
Desvio- Padrão	15,25	0,64	150,45	0,88
Máximo	330	4	1062	3
Mínimo	248	0	0	0
1º Quartil	284,75	0	4	0
3º Quartil	306	0	10	1

Fonte: Autor (2022)

Tabela 55 – Separação: Resultados do Cenário 20 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2185,82	16,27	13,84	10,93	2,90
Moda	2257,24	17,48	13,70	11,23	3,07
Desvio-Padrão	70,07	6,03	0,60	0,35	0,42
Máximo	2375,45	40,68	15,52	11,88	4,42
Mínimo	2037,58	8,41	12,41	10,19	2,20
1º Quartil	2134,69	12,58	13,42	10,67	2,61
3º Quartil	2245,30	17,785	14,20	11,23	3,07

Fonte: Autor (2022)

Analisando-se os resultados das simulações referentes aos cenários 19 e 20, comparados ao cenário 14, pode-se perceber que, com o aumento da separação recomendada, ocorre um aumento no tempo médio das viagens realizadas pelas aeronaves.

Com o aumento da separação horizontal recomendada (cenário 19), gera-se um impacto maior no tempo médio das viagens realizadas, diminuem-se os casos de violações de separação e, ainda, aumenta-se o número de veículos atrasados.

No sentido inverso, com o aumento da separação vertical recomendada (cenário 20), ocorre um aumento do número de violações de separação, principalmente por se colocar mais aeronaves no mesmo nível, já que se diminui o número de níveis em que as aeronaves podem voar. Mas, por outro lado, ocorre uma redução do número de veículos atrasados (com *delay*), um aumento de velocidade média das aeronaves e um maior número de conclusões de viagens, possivelmente pelo resultado da redução no tempo de viagens verticais. Pode-se imaginar, ainda, que com o maior tempo de espera nos níveis horizontais, passam a existir menos conflitos, que poderiam travar as descidas e subidas de aeronaves, o que aumenta ligeiramente a capacidade (eficiência) do espaço.

6.4 RESULTADOS GERAIS – QUANTIDADE DE AERONAVES

Nesta seção são executadas simulações para mais quatro cenários (21 a 24), que são comparados com os resultados obtidos nos cenários 14 (caso 1) e 15 (caso 2), fixados como bases e em que se utilizam trajetórias das aeronaves do tipo “**geral**”, assim como nestes quatro novos cenários. Entre esses cenários variam-se as quantidades máximas de VTOLs do tipo 1 introduzidas no espaço de simulação a cada novo ciclo e mantêm-se as quantidades máximas de VTOLs do tipo 2, conforme apresentado na Tabela 56.

Tabela 56 – Lista de Cenários de Quantidade de Aeronaves

Cenário	Número Máximo de VTOL-1	Número Máximo de VTOL-2
14 (caso 1)	3	1
15 (caso 2)	3	1
21 (caso 1)	6	1
22 (caso 1)	9	1
23 (caso 2)	6	1
24 (caso 2)	9	1

Fonte: Autor (2022)

As tabelas 57 e 58 apresentam os resultados consolidados da execução do cenário 21, enquanto que as tabelas 59 e 60 apresentam os resultados consolidados da execução do cenário 22.

Tabela 57 – Quantidade de VTOL-1: Resultados do Cenário 21 – Caso 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	503,24	0,52	134,19	3,35
Moda	508	0	17	0
Desvio-Padrão	25,67	1,01	297,63	3,42
Máximo	574	4	1215	13
Mínimo	449	0	0	0
1º Quartil	484,75	0	14	0
3º Quartil	519	0	31,50	5

Fonte: Autor (2022)

Tabela 58 – Quantidade de VTOL-1: Resultados do Cenário 21 – Caso 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2577,13	14,60	15,78	12,90	2,88
Moda	-	11,33	15,65	12,78	2,97
Desvio-Padrão	55,017	3,34	0,38	0,28	0,28
Máximo	2714,35	25,90	17,01	13,57	3,85
Mínimo	2357,17	9,39	15,01	11,79	2,34
1º Quartil	2546,48	12,32	15,55	12,74	2,67
3º Quartil	2605,09	16,28	16,01	13,05	3,00

Fonte: Autor (2022)

Tabela 59 – Quantidade de VTOL-1: Resultados do Cenário 22 – Caso 1 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	719,25	1,52	248,31	8,08
Moda	734	0	33	7
Desvio-Padrão	35,79	1,78	464,73	6,38
Máximo	828	8	3252	30
Mínimo	641	0	11	0
1º Quartil	692,75	0	34,75	3
3º Quartil	743	2	204	12

Fonte: Autor (2022)

Tabela 60 – Quantidade de VTOL-1: Resultados do Cenário 22 – Caso 1 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo médio de viagem	Tempo médio de viagem (Vertical)	Tempo médio de viagem (Horizontal)
Média	2584,23	14,66	15,83	12,94	2,89
Moda	2552,89	15,08	15,75	12,80	2,89
Desvio- Padrão	53,07	2,60	0,37	0,26	0,25
Máximo	2740	23,09	16,94	13,73	3,68
Mínimo	2487,22	7,94	15,12	12,44	2,19
1º Quartil	2544,82	13,19	15,61	12,77	2,75
3º Quartil	2614,76	15,76	16,03	13,09	3,02

Fonte: Autor (2022)

As tabelas 61 e 62 apresentam os resultados consolidados da execução do cenário 23, enquanto que as tabelas 63 e 64 apresentam os resultados consolidados da execução do cenário 24.

Tabela 61 – Quantidade de VTOL-1: Resultados do Cenário 23 – Caso 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	513,58	0,42	26,64	1,50
Moda	497	0	6	0
Desvio- Padrão	23,98	1,04	105,38	2,54
Máximo	569	6	743	12
Mínimo	453	0	0	0
1º Quartil	497,75	0	4	0
3º Quartil	529,25	0	11	2

Fonte: Autor (2022)

Tabela 62 – Quantidade de VTOL-1: Resultados do Cenário 23 – Caso 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2587,29	16,40	10,28	7,82	2,45
Moda	-	10,64	10,10	7,76	2,6
Desvio-Padrão	56,92	4,19	0,32	0,17	0,27
Máximo	2740,04	38,41	11,27	8,24	3,51
Mínimo	2437,09	9,32	9,38	7,47	1,90
1º Quartil	2545,35	14,06	10,10	7,71	2,29
3º Quartil	2634,52	17,57	10,44	7,95	2,57

Fonte: Autor (2022)

Tabela 63 – Quantidade de VTOL-1: Resultados do Cenário 24 – Caso 2 (parte 1)

	Viagens Completadas	Abaixo da Separação Mínima	Abaixo da Separação Estipulada	Número de Veículos Atrasados
Média	726,96	0,60	128,25	4,45
Moda	762	0	9	0
Desvio-Padrão	33,97	1,04	275,10	4,43
Máximo	812	4	1262	19
Mínimo	654	0	2	0
1º Quartil	700,25	0	9	1
3º Quartil	752,50	2	25,75	7

Fonte: Autor (2022)

Tabela 64 – Quantidade de VTOL-1: Resultados do Cenário 24 – Caso 2 (parte 2)

	Distância Vertical Média	Distância Horizontal Média	Tempo Médio de Viagem	Tempo Médio de Viagem (Vertical)	Tempo Médio de Viagem (Horizontal)
Média	2584,69	16,10	9,99	7,55	2,44
Moda	-	12,23	10,11	7,48	2,41
Desvio-Padrão	47,80	2,63	0,24	0,15	0,22
Máximo	2691,02	27,82	10,64	8,02	3,16
Mínimo	2471,99	11,40	9,36	7,23	1,91
1º Quartil	2549,69	14,45	9,84	7,45	2,29
3º Quartil	2615,93	17,81	10,11	7,64	2,58

Fonte: Autor (2022)

Pode-se observar que houve um aumento do número de conclusões de viagens com o aumento do número de veículos introduzidos no espaço de simulação a cada ciclo, porém, com o maior congestionamento do espaço aéreo, ocorreu o aumento do número de violações de separação e de atrasos (“*delays*”), mesmo que, nessas condições, não se tenha ainda excedido a capacidade máxima do sistema, que não colapsou. De qualquer forma, considera-se que ocorreram problemas relacionados ao fluxo de aeronaves.

6.5 ANÁLISES GERAIS

Com base nos diversos agrupamentos de simulações realizadas nas seções anteriores, pode-se concluir de uma forma geral o seguinte:

- os cenários com trajetórias “**perpendiculares**” têm maior complexidade do que os cenários com trajetórias “**paralelas**”, pois nos cenários perpendiculares possibilitam uma aeronave cruze o caminho da outra (elas se encontram de maneira que suas trajetórias formam linhas perpendiculares de 90 graus entre si) e, eventualmente, uma delas pode ter que ficar em estado de espera ou reduzir a velocidade. Embora os cenários com trajetórias “**perpendiculares**” possibilitem um aumento da capacidade do espaço aéreo, por permitirem mais alternativas de rotas

para as aeronaves, isso acarreta uma piora geral nos tempos médios de conclusão das viagens e um aumento no número de conflitos gerados;

- os cenários com trajetórias “**gerais**” têm maior complexidade do que os cenários com trajetórias “**paralelas**” e trajetórias “**perpendiculares**”, pois apresentam múltiplos eixos de navegação num nível do espaço aéreo. Mas, apesar disso, proporcionam uma melhoria em relação aos cenários com trajetórias “**paralelas**” e trajetórias “**perpendiculares**”, pois as aeronaves conseguem desenvolver maiores velocidades, tendo mais alternativas para se deslocarem num dado nível do espaço aéreo. Assim, os tempos médios de conclusão das viagens diminuem em relação aos outros cenários e ocorre, também, uma diminuição no número de conflitos gerados.
- todos os tipos de cenários, com trajetórias “**gerais**”, “**paralelas**” e “**perpendiculares**” foram afetados de forma negativa com a introdução de condições climáticas adversas durante as simulações, o que ocorreu por meio da introdução de CBs de dois tipos, pois a presença dessas formações causa um impacto na capacidade do espaço aéreo em todos os cenários apresentados, mesmo em uma condição em que não exista muitas aeronaves navegando simultaneamente no espaço simulado;
- o aumento da separação horizontal acarretou uma diminuição do número de conflitos gerados, porém fez com que houvesse uma redução considerável da capacidade do espaço aéreo simulado; por outro lado, o aumento da separação vertical sem o aumento também da separação horizontal, gerou um aumento do número de conflitos porque diminuiu-se, conseqüentemente, o número de níveis de voo, fazendo com que mais aeronaves fiquem em um mesmo nível;
- a variação do número de veículos concluindo viagens, por período de simulação, depende mais fortemente do número de veículos que entram no espaço aéreo simulado por ciclo de simulação do que da variação de outros parâmetros, como a velocidade máxima desenvolvida pelas aeronaves ou separação de aeronaves, pois pode ocorrer de haver uma melhora individual do desempenho de cada aeronave (redução do tempo médio de viagens), mas não ocorrer uma melhoria no fluxo do tráfego, por haver poucas aeronaves navegando simultaneamente;

- o aumento da quantidade de aeronaves navegando simultaneamente pode acarretar uma degradação do espaço aéreo simulado, pois muitas aeronaves próximas tendem a gerar um aumento no número de conflitos no espaço aéreo e também um maior número de esperas (“*delays*”). O aumento do número de veículos atrasados mostra que existem regiões de congestionamento, o que, em geral, implica a geração de um maior número de conflitos. Contabilizar o número de atrasos é, portanto, tão relevante quanto a totalização do número de conflitos ocorridos, para a avaliação da capacidade do espaço aéreo.

7. CONCLUSÃO

Este trabalho foi iniciado com uma revisão do estado da arte quanto ao uso de sistemas multiagentes aplicados ao UAM, relacionando diferentes trabalhos e sistemas existentes. A partir disso, pôde-se verificar os diversos problemas (e suas complexidades) que surgiram a partir da concepção do UAM e da navegação de eVTOLs em uma área urbana, bem como outros trabalhos que estão em desenvolvimento com o objetivo de propor soluções para esses delicados e importantes problemas.

Isso posto, foi proposto e projetado um modelo multiagentes, conforme apresentado nesse trabalho, a partir de regras existentes tanto em outros trabalhos como na aviação tradicional. Com essa análise, é possível concluir que o modelo aqui proposto faz sentido quando aplicado a um cenário real, assim como também é possível verificar que os dados gerados estão conforme o esperado. Em decorrência das simulações realizadas, pôde-se tirar conclusões como resultado de uma análise maior baseada na construção de hipóteses, variando-se, para isso, os parâmetros de entrada. Isso não só valida a utilidade da simulação como ferramenta para validar o modelo, como também mostra a utilidade dos dados gerados para se tirar conclusões e fazer hipóteses.

Dessa forma, pôde-se executar simulações do modelo desenvolvido durante esse trabalho e, a partir disso, realiza ajustes nos parâmetros de entrada para, então, se obter resultados para serem analisados. Com isso, para diferentes parâmetros de entrada utilizados, observaram-se os impactos que acontecem em decorrência disso na segurança operacional do tráfego aéreo no fluxo de aeronaves.

Pôde-se também perceber o potencial do modelo criado para gerar um grande número de cenários relevantes, utilizando diferentes parâmetros de entrada, para que sejam realizadas validações dos casos selecionados.

O modelo permite que sejam criados múltiplos cenários, utilizando múltiplas aeronaves, sem risco operacional para pessoas ou equipamentos, a um custo significativamente menor do que seria caso equipamentos reais fossem utilizados. Isso é útil, principalmente, nos cenários considerando a existência de CBs, que seriam situações difíceis de serem geradas e reproduzidas no mundo real.

Como as execuções de simulações considerando os parâmetros de entrada são geradas de maneira estocástica, mesmo com a utilização dos mesmos valores

para execuções distintas, pode-se obter resultados com certa variabilidade de valores, o que permite se fazer uma análise estatística com a distribuição dos dados obtidos, para então se comparar com mais propriedade os resultados de diferentes cenários.

7.1 CONTRIBUIÇÕES

Durante o desenvolvimento desse trabalho, as seguintes contribuições foram geradas:

- **Proposição de um modelo multiagente:** durante esse trabalho, pôde-se construir um modelo multiagente e, a partir disso, verificar seu potencial para a modelagem de sistemas utilizando eVTOLs no contexto do sistema de UAM. Então, a partir das regras previamente definidas, foi possível construir este modelo, que pode ser utilizado na área aeronáutica ou ser ampliado para outros usos e em outras áreas;
- **Revisão de regras do UAM:** durante esse trabalho, foi feita uma revisão de diversas regras da literatura aplicadas ao tráfego UAM e ao tráfego aéreo tradicional, permitindo a validação e o uso dessas regras em um ambiente simulado;
- **Execução e validação do modelo:** após o desenvolvimento do modelo foi possível executar uma série de ajustes e calibração nesse modelo. Também foram executadas simulações do modelo utilizando diversos valores dos parâmetros de entrada, que levaram à adição de novos parâmetros, resultando no refinamento do modelo final apresentado nesse trabalho;
- **Resultados gerados pelo modelo:** após a execução desse modelo considerando diversos cenários, os resultados obtidos permitiram se estabelecer conclusões, considerando a interação de múltiplos parâmetros.

7.2 TRABALHOS FUTUROS

Durante o desenvolvimento desse trabalho, vislumbrou-se a possibilidade de desenvolvimento de melhorias no modelo concebido, ou o desenvolvimento de

trabalhos continuando ou estendendo esta abordagem. Tais possibilidades são listadas seguir:

- **Ampliação do modelo:** além do modelo no seu estado atual apresentado, pode-se fazer uma atualização desse modelo, adicionando-se mais parâmetros de entrada, como novos tipos de veículos ou novos tipos de cenários de simulação, o que permitiria a realização de novas simulações e a conclusão de novas hipóteses;
- **Validação do modelo com diferentes regras do UAM:** além do modelo utilizado nesse trabalho desenvolvido para o UAM, é possível utilizar diferentes ideias, regras e tipos de aeronaves, com a sua combinação com o tráfego aéreo tradicional, permitindo, assim, estender esse modelo para a criação de novos modelos mais complexos e abrangentes, com a utilização das ideias contidas nesse trabalho;
- **Uso de modelos multiagentes em outras áreas:** pode-se verificar o uso de sistemas multiagentes como eficientes em diversas áreas, como no UAM dentro da área de aviação. Isso pode ser útil para abrir caminho para outras pesquisas que envolvam simulações, com sistemas multiagentes, com diversos propósitos, ainda sobre o UAM;
- **Outras aplicações:** pode-se obter uma aplicação do modelo aqui desenvolvido em cenários reais e também a utilização dos dados gerados para diferentes análises em outros tipos de trabalhos, como a modelagem em redes de Markov para medir complexidade do espaço aéreo, conforme trabalho descrito em (Baum, 2021).

7.3 CONSIDERAÇÕES FINAIS

O modelo multiagente desenvolvido neste trabalho de pesquisa pôde ser validado por meio de múltiplas simulações computacionais, devendo ser destacado que as análises dos resultados não são necessariamente triviais, já que a presença de um veículo adicional no espaço aéreo, por exemplo, é capaz de impactar o fluxo de outras aeronaves, e de maneira sucessiva, gerar impactos em todo o sistema de tráfego aéreo. Com isso, é necessário se avaliar os resultados gerados de maneira global – o que foi feito neste trabalho de pesquisa –, apesar de eventuais parâmetros tratarem individualmente de uma dada aeronave.

Para isso, como o modelo desenvolvido realiza simulações de forma estocástica, é necessário se criar diversos cenários estáticos para validar o funcionamento básico do modelo e, posteriormente, acompanhar o funcionamento de diversos cenários estáticos até se ganhar confiança no modelo, antes de se executar múltiplas simulações e de se fazer a análise estatística nos resultados.

Pode-se, também, verificar o potencial do modelo multiagente concebido na simulação de cenários complexos e, com isso, vislumbrar-se a construção de modelos ou a extensão do modelo para se obter efeitos benéficos, como poder se validar a segurança de um dado espaço aéreo sem a necessidade de se colocar fisicamente aeronaves navegando nesse espaço.

REFERÊNCIAS

ANAC. Orientações para usuários de Drones. [S.l.], 2017. Disponível em: <https://web.archive.org/web/20190214154533/http://www.anac.gov.br/assuntos/paginas-tematicas/drones/orientacoes_para_usuarios.pdf>. Acesso em 14/02/2019.

ASHISH, M.; et al. "Autonomous Payload Delivery using Hybrid VTOL UAVs for Community Emergency Response," 2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp. 1-6, doi: 10.1109/CONECCT50063.2020.9198490. 2020.

BALACHANDRAN, S., National Institute of Aerospace, Hampton, VA, Anthony Narkawicz, César Muñoz and Maria Consiglio, NASA Langley. A Path Planning Algorithm to Enable Well-Clear Low Altitude UAS Operation Beyond Visual Line of Sight. Twelfth USA/Europe Air Traffic Management Research and Development Seminar (ATM 2017), Seattle, WA. June, 2017.

BALLVE, M. DRONES: Commercial Drones Are Becoming a Reality, With Huge Impacts for Many Industries. 2014. Disponível em: <<https://web.archive.org/web/20190212174422/https://www.businessinsider.com/drones-navigating-toward-commercial-applications-2014-5>>. Acesso em 21/09/2019.

BAUM, D. M. Airspace Capacity Artificial Intelligence Model in UAM Environment Based on The Airspace Complexity. 2021. 199 f. Tese (Doutorado em Engenharia Elétrica) - Universidade de São Paulo. 2021.

BAURANOV, A.; RAKAS, J., Designing airspace for urban air mobility: A review of concepts and approaches, Progress in Aerospace Sciences, Volume 125, ISSN 0376-0421, <https://doi.org/10.1016/j.paerosci.2021.100726>. 2021.

BBC. Uber sells off flying taxi unit. 2021. Disponível em: <<http://web.archive.org/web/20220619144250/https://www.bbc.com/news/business-55241172>>. Acesso em 19/06/2022.

BEARD, R., MCLAIN, T. Small unmanned aircraft: Theory and Practice. Princeton University Press. 2012.

BESADA-PORTAS, EVA; LA TORRE, LUIS DE; MORENO, ALEJANDRO; RISCO-MARTÍN, JOSÉ L.. On the performance comparison of multi-objective evolutionary UAV path planners, Information Sciences, Volume 238, 2013, Pages 111-125, ISSN 0020-0255, doi: 10.1016/j.ins.2013.02.022.

BLANK, P.; KIRrane, S.; SPIEKERMANN, S. Privacy-aware restricted areas for unmanned aerial systems. IEEE Security Privacy, v. 16, n. 2, p. 70–79, March 2018. ISSN 1540-7993.

BO, N. et al. A hierarchical optimization strategy of trajectory planning for multi-uavs. In: 2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). [S.l.: s.n.], 2017. v. 1, p. 294–298.

BRADSHAW, J. M. Software Agents. ISBN-13: 978-0262522342. AAI. 1997.

CHAVES, Á. N. Proposta de modelo de veículos aéreos não tripulados (VANTS) cooperativos aplicados a operações de busca. 149 p. Tese (Doutorado), 2013.

CHEN, Y. et al. Trajectory optimization of multiple quad-rotor UAVs in collaborative assembling task. Chinese Journal of Aeronautics, v. 29, n. 1, p. 184 – 201, 2016. ISSN 1000-9361. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S100093611500237X>>.

COELHO, B. N. et al. A multi-objective green UAV routing problem. Computers Operations Research, v. 88, p. 306 – 315, 2017. ISSN 0305-0548. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0305054817301028>>.

COHEN, A. P.; SHAHEEN, S. A.; FARRAR, E. M.. "Urban Air Mobility: History, Ecosystem, Market Potential, and Challenges," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 9, pp. 6074-6087, Sept. 2021, doi: 10.1109/TITS.2021.3082767.

CORRÊA, M. A. Modelo de Veículos Aéreos não Tripulados Baseado em Sistemas Multi-Agentes. 90 p. Tese (Doutorado) — Universidade de São Paulo (USP), 2008.

DECEA. Carta Aérea Rio-São Paulo. <https://geoaisweb.decea.mil.br/#> . Acessado em: 04/08/2022.

DEMETRIOU, D. Flying cars set for takeoff at Tokyo Olympics in 2020. 2017. Disponível em: <<https://web.archive.org/web/20190308122212/https://www.telegraph.co.uk/technology/2017/05/17/flying-cars-set-takeoff-tokyo-olympics-2020/>>. Acesso em 08/03/2019.

DESJARDINS, J. Amazon and UPS are betting big on drone delivery. 2018. Disponível em: <<https://web.archive.org/web/20190209014840/https://www.businessinsider.com/amazon-and-ups-are-betting-big-on-drone-delivery-2018-3/>>. Acesso em 09/02/2019.

ÉPOCA NEGÓCIOS. Drones dão show na cerimônia de abertura das Olimpíadas de Tóquio 2020. 2021. Disponível em: <<http://web.archive.org/web/20220619140221/https://epocanegocios.globo.com/Tecnologia/noticia/2021/07/drones-dao-show-na-cerimonia-de-abertura-das-olimpiadas-de-toquio-2020.html>> . Acesso em 19/06/2022.

FAA. Pilot's handbook of Aeronautical Knowledge. FAA-H-8083-25B. 2016.

FAA. UAM Concept of Operations v1.0. https://nari.arc.nasa.gov/sites/default/files/attachments/UAM_ConOps_v1.0.pdf . June, 2020.

FAHMANI, L.; GARFAP J.; BOUKHDIR K.; BENHADOU S.; MEDROMI H.. "Unmanned Aerial Vehicles Inspection for Overhead High Voltage Transmission Lines," 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), 2020, pp. 1-7, doi: 10.1109/IRASET48871.2020.9092141

FERRARE, F. D.; BAUM, D. MOREIRA; ALMEIDA JÚNIOR, J. R. DE; CAMARGO JÚNIOR, J. B.; CUGNASCA, P. S.. "Scenarios for the Use of eVTOLs Using Multiagent Systems With Netlogo: Comparison Of Parameters And The Impact On UAM," 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), 2021, pp. 1-7, doi: 10.1109/DASC52595.2021.9594507.

FERRARE, F. D.; BAUM, D. MOREIRA; ALMEIDA JÚNIOR, J. R. DE; CAMARGO JÚNIOR, J. B.; CUGNASCA, P. S.. Urban Air Mobility (UAM): A Model Proposal based on Agents using Netlogo. In Proceedings of the 11th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH, ISBN 978-989-758-528-9; ISSN 2184-2841, pages 352-359. DOI: 10.5220/0010557203520359. 2021.

FLIGHTCREWGUIDE. Semicircular Cruising Level System (ICAO). <https://web.archive.org/web/20220804235827/https://flightcrewguide.com/wiki/rules-regulations/semicircular-cruising-level-system-icao/>. Acessado em: 04/08/2022.

G1. Dois são presos por usar drones em área da tragédia em Brumadinho. 2019. Disponível em: <<https://web.archive.org/web/20190214162630/https://g1.globo.com/mg/minas-gerais/noticia/2019/02/04/dois-sao-presos-por-usar-drones-em-area-da-tragedia-da-vale-em-brumadinho.ghtml>>. Acesso em 14/02/2019.

G1. Carro voador faz voo de 4 minutos em teste no Japão. 2020. Disponível em: <<https://web.archive.org/web/20210521164132/https://autoesporte.globo.com/videos/noticia/2020/08/carro-voador-faz-voo-de-4-minutos-em-teste-no-japao-assista.ghtml>>. Acesso em 21/05/2021.

GIYENKO, A.; CHO, Y. I.. "Intelligent UAV in smart cities using IoT," 2016 16th International Conference on Control, Automation and Systems (ICCAS), Gyeongju, 2016, pp. 207-210, doi: 10.1109/ICCAS.2016.7832322.

GLAAB, LOUIS J., et al., NASA Langley Research Center, Donald E. Harper, Analytical Mechanics Associates, Inc. Hampton, VA. https://utm.arc.nasa.gov/docs/2018-Glaab_NASA-TM-219816.pdf, NASA Technical Memorandum, April, 2018.

GRIMM, VOLKER, et al.. "The ODD Protocol for Describing Agent-Based and Other Simulation Models: A Second Update to Improve Clarity, Replication, and Structural Realism", Journal of Artificial Societies and Social Simulation 23 (2) 7 <<http://jasss.soc.surrey.ac.uk/23/2/7.html>>. doi: 10.18564/jasss.4259. 2020.

GUARDIAN. Heathrow: police investigate after drone sighting disrupts flights. 2019. Disponível em:

<<https://web.archive.org/web/20190214162814/https://www.theguardian.com/uk-news/2019/jan/08/heathrow-airport-departures-suspended-after-drone-sighting>>. Acesso em 14/02/2019.

HAWKINS, A. J. Uber narrows its search for international city to host 'flying táxis'. 2018. Disponível em: <<https://web.archive.org/web/20190211172037/https://www.theverge.com/2018/8/30/17795588/uber-elevate-flying-car-international-city-search-drone>>. Acesso em 11/02/2019.

HOLLISTER, S. Drone sighting briefly halts flights to Newark International Airport. 2019. Disponível em: <<https://web.archive.org/web/20190212174557/https://www.theverge.com/2019/1/22/18193593/drone-sighting-briefly-halts-flights-to-newark-international-airport>>. Acesso em 21/02/2019.

HOMOLA, J. et al. UAS traffic management (UTM) simulation capabilities and laboratory environment. AIAA/IEEE Digital Avionics Systems Conference - Proceedings, v. 2016-December, p. 1–7, 2016. ISSN 21557209.

IBGE. Cidade de São Paulo. Em <https://cidades.ibge.gov.br/brasil/sp/sao-paulo/panorama> . Acessado em 02/11/2022, 2022.

IG. Carro voador da Embraer: ações da Eve despencam no 1º dia nos EUA. 2022. Disponível em: <<http://web.archive.org/web/20220619143555/https://economia.ig.com.br/2022-05-11/carro-voador-eve-acoes-despencam.html>>. Acesso em 19/06/2022.

JAIN A.; et al., "Baseline procedure for conceptual designing of an eVTOL for Urban Air Mobility," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2020, pp. 1173-1178, doi: 10.1109/ICECA49313.2020.9297649.

JOSHI, D. What are drones good for? Common commercial applications of drones in agriculture, business and the military. 2017. Disponível em: <<https://web.archive.org/web/20190211173359/https://www.businessinsider.com/commercial-drone-uses-agriculture-business-military-2017-8>>. Acesso em 11/02/2019.

KIM, S. Multicopter configurations of an unmanned mini aircraft. 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC). doi=10.1109/ICNSC.2018.8361331. 2018.

LEI, G. et al. Multi-agent path planning for unmanned aerial vehicle based on threats analysis. 2011 3rd International Workshop on Intelligent Systems and Applications, ISA 2011 - Proceedings, n. 1, p. 1–4, 2011.

LUNDBERG, J.; K. L. PALMERIUS AND B. JOSEFSSON, "Urban Air Traffic Management (UTM) Implementation In: Cities - Sampled Side-Effects," 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), pp. 1-7. doi: 10.1109/DASC.2018.8569869. London, 2018.

MACAL, C M; NORTH, M J. Tutorial on agent-based modeling and simulation, *Journal of Simulation*, 4:3, 151-162, DOI: 10.1057/jos.2010.3. 2010.

MAGET,C.; Gutmann, S.; Bogenberger, K. "Model-based Evaluations Combining Autonomous Cars and a Large-scale Passenger Drone Service: The Bavarian Case Study," 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pp. 1-6, doi: 10.1109/ITSC45102.2020.9294183. 2020.

METRÓPOLES. É oficial! Pira Olímpica é acesa em Tóquio por Naomi Osaka. 2021. Disponível em: <<http://web.archive.org/web/20220619140719/https://www.metropoles.com/esportes/toquio-2020/sem-publico-abertura-das-olimpiadas-reune-desfiles-e-protestos>>. Acesso em 19/06/2022.

MINHAPAIXAOEAVIAOEAVIACAO. Disponível em: <<http://minhapaixaoeaviaoeaviacao.blogspot.com/2015/08/cumulus-nimbus-oque-isso.html>>. Acesso em 01/05/2023.

MONTAMBAULT, S.; Beaudry, J.; Toussaint, K.; Pouliot, N. "On the application of VTOL UAVs to the inspection of power utility assets," 2010 1st International Conference on Applied Robotics for the Power Industry, pp. 1-7, doi: 10.1109/CARPI.2010.5624443. 2010.

NASA UAM. site: <https://utm.arc.nasa.gov/index.shtml> (consultado em Jan/2021), 2021.

NETO, E. C. P.; Baum, D. M.; Almeida, J. R.; Camargo, J. B.; Cugasca, P. S. UAS in the Airspace: A Review on Integration, Simulation, Optimization, and Open Challenges. <https://doi.org/10.48550/arxiv.2211.15330> . ArXiv. 2022.

REICHE, C.; McGillen, C.; Siegel, J.; Brody; F. "Are We Ready to Weather Urban Air Mobility (UAM)?" 2019 Integrated Communications, Navigation and Surveillance Conference (ICNS), pp. 1-7, doi: 10.1109/ICNSURV.2019.8735297. 2019.

ROCKET CUMULONIMBUS. Disponível em: <<https://g3i5r4x7.rocketcdn.me/wp-content/uploads/2020/09/cumulonimbus-o-que-e-conceito-tipos-e-caracteristicas.jpg>> . Acesso em 01/05/2023.

RUSSELL, S. J.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. 3a edição. Editora Prentice Hall. 2009.

O'KANE, S. Drones are already being confiscated near the Super Bowl. 2019. Disponível em: <<https://web.archive.org/web/20190212174623/https://www.theverge.com/2019/2/11/8207081/super-bowl-2019-drones-atlanta-fbi>>. Acesso em 12/02/2019.

PANAIT, L.; LUKE, S.. *Cooperative Multi-Agent Learning: The State of the Art*. *Autonomous Agents and Multi-Agent Systems*. 11 (3): 387–434. CiteSeerX 10.1.1.307.6671. doi:10.1007/s10458-005-2631-2. 2005.

PAPERT, S. MINDSTORMS: Children, Computers, and Powerful Ideas. Basic Books, Inc., Publishers. New York. 1980.

PREFEITURA DE SP. Cap. 2 - CARACTERÍSTICAS GERAIS DO MUNICÍPIO. https://www.prefeitura.sp.gov.br/cidade/secretarias/upload/arquivos/secretarias/meio_ambiente/projetos_acoes/0004/capitulo2.pdf . Acessado em 02/11/2022, 2022.

SHAKHATREH, H. et al. Unmanned Aerial Vehicles: A Survey on Civil Applications and Key Research Challenges. arXiv e-prints, p. arXiv:1805.00881, Apr 2018.

SILVA, R. D.; RAJASINGHEGE, S. Optimal desired trajectories of UAVs in private UAV networks. In: 2018 International Conference on Advanced Technologies for Communications (ATC). [S.l.: s.n.], 2018. p. 310–314. ISSN 2162-1039.

SILVEIRA, D. T.; CÓRDOVA, F. P. A pesquisa científica. Porto Alegre: Editora da UFRGS, 2009. 109 p. ISBN 9788538600718.

STAFF, V. Idiots with drones shut down the UK's second largest airport again. 2018. Disponível em: <<https://web.archive.org/web/20190211075406/https://www.theverge.com/2018/12/20/18149819/london-gatwick-airport-drone-shutdown-reports>>. Acesso em 11/02/2019.

THIPPHAVONG, DAVID P.; APAZA, RAFAEL D.; BARMORE, BRYAN E.; ET AL. Urban Air Mobility Airspace Integration Concepts and Considerations. <<https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20180005218.pdf>>. 2019.

THIELE, J. C.; KURTH, W.; GRIMM, V. 'Facilitating Parameter Estimation and Sensitivity Analysis of Agent-Based Models: A Cookbook Using NetLogo and 'R'' Journal of Artificial Societies and Social Simulation 17 (3) 11 <<http://jasss.soc.surrey.ac.uk/17/3/11.html>>. doi: 10.18564/jasss.2503. 2014.

WAKEFIELD, J. Dubai tests drone taxi service. 2017. Disponível em: <<https://web.archive.org/web/20190308122328/https://www.bbc.com/news/technology-41399406>>. Acesso em 08/03/2019.

WAZLAWICK, R. S. Metodologia de pesquisa para ciência da computação. Rio de Janeiro: Elsevier, 2009. p. 159. ISBN 978-85352-3522-7.

WEI, Z.; LI, J. Design of Air Traffic Control Operation System Using Multi Agent Technology and Simulation. The Open Fuels & Energy Science Journal, n. 8, p. 368–374, 2015.

WEN, N. et al. Online UAV path planning in uncertain and hostile environments. International Journal of Machine Learning and Cybernetics, v. 8, n. 2, p. 469–487, Apr 2017. ISSN 1868-808X. Disponível em: <<https://doi.org/10.1007/s13042-015-0339-4>>.

WILENSKY, U.. NetLogo. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. 1999.

WILENSKY, U. Netlogo Airplane Landing Example 3D. <http://ccl.northwestern.edu/netlogo/models/AirplaneLandingExample3D>. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. 2011.

WILENSKY, U; RAND, W. Introduction to agent-based modeling modeling natural, social, and engineered complex systems with NetLogo. Cambridge, Mass MIT Press. 2015.

WING, D. J.; COTTON, W. B. For Spacious Skies: Self-Separation with "Autonomous Flight Rules" in US Domestic Airspace. 11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference; September 20, 2011 - September 22, 2011; Virginia Beach, VA; United States. 2011.

WOOLDRIDGE, M. J., An introduction to multiagent systems. ISBN 0-471-4969 I-X. Editora Wiley. 2002.

WOOLDRIDGE, M. J., Reasoning about Rational Agents. Editora MIT. 2000.

XIE, J. & LIU, C.-C. Multi-agent systems and their applications, Journal of International Council on Electrical Engineering, 7:1, 188-197, DOI: 10.1080/22348972.2017.1348890. 2017.

YANG, Z.; FANG, Z.; LI, P. Bio-inspired collision-free 4d trajectory generation for uavs using tau strategy. Journal of Bionic Engineering, v. 13, n. 1, p. 84 – 97, 2016. ISSN 1672-6529. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1672652914601621>>.

ZHAO, P. et al. Optimal trajectory planning of drones for 3d mobile sensing. In: 2018 IEEE Global Communications Conference (GLOBECOM). [S.l.: s.n.], 2018. p. 1–6. ISSN 2576-6813

APÊNDICE A– Descrição detalhada do Modelo

Temos aqui uma descrição dos itens que existem no modelo:

A – Botões de Execução do Modelo:

- **Setup**: permite limpar os resultados anteriores de uma simulação para se iniciar uma nova simulação.
- **GO (com loop)**: executa a simulação do modelo até atingir uma condição de parada previamente definida.
- **GO**: executa um minuto de simulação a cada pressionar desse botão; trata-se de um recurso útil para a depuração de uma simulação, podendo-se avaliar a cada passo o resultado parcial da simulação.

B – Parâmetros de Entrada do Simulador:

Scenario:

- Permite selecionar o tipo de cenário a ser simulado:
 - **Paralelo**: cenário de simulação no qual as rotas realizadas por cada aeronave são paralelas, sem cruzamento entre veículos.
 - **Perpendicular**: cenário de simulação no qual as rotas realizadas por cada aeronave podem se cruzar, sempre perpendicularmente (horizontal ou verticalmente).
 - **Geral**: cenário de simulação no qual as rotas realizadas por cada aeronave podem ou não se cruzar, e quando se cruzam pode ser em qualquer direção (horizontal, vertical ou diagonalmente).

Conclusion_type:

- **Tempo**: execução do modelo termina após um período de tempo previamente selecionado.
- **Ocorrência de uma colisão**: execução do modelo termina após a ocorrência de uma colisão entre aeronaves.

Time Elapsed:

- **Tempo**: parâmetro de tempo que determina o término da execução do modelo quando configurado para terminar após um determinado período de tempo, conforme o “Conclusion_type”.

Number of generation per minute:

- **NumberMaxVTOL1:** número máximo de aeronaves do tipo VTOL1 a serem adicionadas no sistema a cada ciclo (TimeGenerationInterval).
- **NumberMaxVTOL2:** número máximo de aeronaves do tipo VTOL2 a serem adicionadas no sistema a cada ciclo (TimeGenerationInterval).
- **TimeGenerationInterval:** intervalo de tempo (chamado no sistema de “ciclo”) durante o qual serão inseridos, de forma estocástica, um número máximo de VTOL do tipo 1 e um número máximo de VTOL do tipo 2 na simulação, respectivamente, NumberMaxVTOL1 e NumberMaxVTOL2.
- **LimitPlanes:** parâmetro voltado para a depuração do simulador; quando ativado (On), ao se atingir o LimitPlanesQnt, nenhuma nova aeronave é inserida no sistema a cada ciclo, a menos que aeronaves pousem no seu destino; quando desativado (Off), não existe esta restrição.
- **LimitPlanesQnt:** quantidade máxima de aeronaves navegando simultaneamente no simulador, caso o parâmetro LimitPlanes esteja ativado (On).
- **SizeWord:**
 - **15x15(NM):** limite do espaço de simulação restrito a 15NM x 15 NM.
 - **30x30(NM):** limite do espaço de simulação restrito a 30NM x 30 NM.

Number of CBs:

- **CB:** número de CBs do tipo 1 (formato mais vertical) que serão gerados dentro do espaço de simulação, sendo que, quando igual a 0, a funcionalidade está desativada.
- **CB2:** número de CBs do tipo 2 (formato mais horizontal) que serão gerados dentro do espaço de simulação, sendo que, quando igual a 0, a funcionalidade está desativada.

Safety Parameters:

- **Sep_horizontal:** separação mínima regulamentada entre aeronaves no plano horizontal.
- **Sep_vertical:** separação mínima regulamentada entre aeronaves no plano vertical.

Speed parameters:

- **VerticalSpeed_VTOL:** velocidade máxima vertical do VTOL do tipo 1.
- **VerticalSpeed_VTOL2:** velocidade máxima vertical do VTOL do tipo 2.

- **SpeedHorizontal_VTOL:** velocidade máxima horizontal do VTOL do tipo 1.
- **SpeedHorizontal_VTOL2:** velocidade máxima horizontal do VTOL do tipo 2.

C – Parâmetros de controle do simulador:

Batch execution:

- **Num_Batch:** número de execuções estocásticas, com os mesmos parâmetros, que gerarão resultados em um arquivo do tipo CSV.
- **RunBatch:** inicia a execução do *batch* de simulação.
- **Stop_Var:** trata de um sinalizador que, quando ativado (On), interrompe a simulação do tipo *batch* antes que ela atinja o seu final; quando desativado (Off), permite a simulação do tipo *batch*.

Debug Params:

- Diversos parâmetros que permitem ativar ou desativar funções estocásticas de execução do modelo; são úteis para o teste do modelo, gerando resultados determinísticos.

D – Parâmetros de saída do simulador:

Total Planes: total de aeronaves que estão navegando no ambiente de simulação.

Total VTOLs: total de aeronaves do tipo VTOL1 que estão navegando no ambiente de simulação.

Total VTOLs2: total de aeronaves do tipo VTOL2 que estão navegando no ambiente de simulação.

Number of Vehicles: quantidade de aeronaves que concluíram a rota entre a decolagem da origem e o pouso no destino.

Below Minimum Separation: quantidade de eventos de violação de separação entre duas aeronaves que ocorreram durante uma simulação. Essa violação de separação ocorre quando uma aeronave está parada dentro de uma quadrícula acima de um ponto de decolagem, e uma outra aeronave decola e adentra essa mesma quadrícula antes que a aeronave anterior saia dessa quadrícula. Este tipo de conflito somente ocorre quando o espaço aéreo está congestionado. Nesse caso, pode ter ocorrido uma colisão.

Below Stipulated Separation: quantidade de eventos de violação de separação regulamentar entre duas aeronaves que ocorreram durante uma simulação. Essa violação de separação ocorre quando uma aeronave está muito próxima de uma outra aeronave, menor do que o valor mínimo definido com parâmetro (**Sep_horizontal** ou **Sep_vertical**), porém em quadrículas separadas. Neste caso, com certeza não ocorreu a colisão; apenas o espaço aéreo está congestionado a ponto de as aeronaves estarem muito próximas umas das outras.

Delay: quantidade de eventos durante a simulação de aeronaves que estão dentro de uma quadrícula há mais de 10 minutos e que não conseguem ir para nenhuma direção (aeronave em estado de espera).

Input/Output Ratio: razão entre aeronaves que estão entrando e estão saindo do ambiente de simulação. Quando maior do que 1 durante longos períodos de tempo, significa que o ambiente de simulação está entrando em congestionamento.

AVG Time VTOL: tempo médio da rota de um VTOL do tipo 1.

AVG Time VTOL2: tempo médio da rota de um VTOL do tipo 2.

AVG Time: tempo médio da rota de um VTOL de qualquer tipo, 1 ou 2.

Vertical VTOL: tempo médio de deslocamento vertical (processo de decolagem e de pouso) de um VTOL do tipo 1.

Vertical VTOL2: tempo médio de deslocamento vertical (processo de decolagem e de pouso) de um VTOL do tipo 2.

AVG Vertical: tempo médio de deslocamento vertical (processo de decolagem e de pouso) de um VTOL de qualquer tipo, 1 ou 2.

AVG Distance Vertical: distância média percorrida pelas aeronaves durante o deslocamento vertical (processo de decolagem e de pouso) de VTOLs de qualquer tipo, 1 ou 2.

Horiz. VTOL: tempo médio de deslocamento horizontal de um VTOL do tipo 1.

Horiz. VTOL2: tempo médio de deslocamento horizontal de um VTOL do tipo 2.

AVG Horiz.: tempo médio de deslocamento horizontal de um VTOL de qualquer tipo, 1 ou 2.

AVG Distance Horizontal: distância média percorrida pelas aeronaves durante o deslocamento horizontal de VTOLs de qualquer tipo, 1 ou 2.

E – Gráficos de saída do simulador:

Total flying: evolução, ao longo do tempo de simulação, do total de aeronaves que estão navegando no ambiente de simulação. Esse número, ao longo do tempo, cresce quando aeronaves decolam e decresce quando aeronaves pousam.

Number of Vehicles: evolução, ao longo do tempo de simulação, do total de aeronaves que concluíram a sua missão entre a decolagem da origem e o pouso no destino. Esse número só cresce ao longo do tempo.

Separation: evolução, ao longo do tempo de simulação, do total de violações de separação (mínima e regulamentar) e de *delay*. Esse número só cresce ao longo do tempo.

Input / Output: evolução, ao longo do tempo de simulação, da razão entre aeronaves que estão entrando e estão saindo do ambiente de simulação. Quando esse número fica maior do que 1 durante longos períodos de tempo, significa que o ambiente de simulação está entrando em congestionamento.

APÊNDICE B – Código Fonte do Modelo Netlogo Desenvolvido

Nas seções a seguir é apresentado o código fonte do modelo desenvolvido na linguagem Logo. O código está dividido em várias subseções do apêndice, para facilitar a leitura e referência.

Nesse código apresentado foram mantidos os textos existentes nos comentários. Como referência da linguagem Logo, qualquer linha após o texto “;” (dois pontos e vírgulas) se torna um comentário. Para manter visíveis os comentários e diferenciá-los do programa, os comentários foram colocados na cor cinza, enquanto que as palavras chaves do Netlogo foram mantidas na cor azul claro (segundo o padrão da IDE do Netlogo).

O programa foi dividido em 9 subseções para maior clareza, conforme apresentado a seguir:

- Subseção 1: encontra-se a “Definição das entidades”, local em que se define quais componentes do Netlogo são utilizados, os agentes e as suas variáveis internas;
- Subseção 2: existem algumas definições gerais usadas pelo modelo;
- Subseção 3: encontra-se o *setup* das diversas variáveis utilizadas na execução do modelo;
- Subseção 4: são apresentadas as configurações de CBs;
- Subseção 5: existe a lógica de Bloqueio de quadrículas e desvios e lógicas relacionadas;
- Subseção 6: encontra-se a lógica que controla a execução em si do modelo;
- Subseção 7: são geradas e configuradas as aeronaves;
- Subseção 8: encontra-se a lógica que coordena o movimento das aeronaves, incluindo subidas, descidas e desvios;
- Subseção 9: encontra-se a lógica necessária para a execução em *batch* das simulações, a geração de relatórios no formato de arquivo CSV e seu armazenamento no *filesystem* na máquina que executa o modelo.

Apêndice B.1 DEFINIÇÃO DAS ENTIDADES

```

.....
;;
;; (C) Copyright 2020
;; Felipe Desiglo Ferrare <desiglo1@hotmail.com>
;;
;;
.....

extensions [ csv ]      ;; CSV file manipulation library used by Batch functions

breed [VTOLs VTOL]    ;; Creating two types of planes
breed [UASs UAS]

.....
;; Entities definitions
;
.....

VTOLs-own [
  destX destY destZ      ;; Destiny coordinates
  OrigX OrigY OrigZ      ;; Original Location
  colision                ;; Collision Flag
  angV                    ;; AngV is used when the diagonal running
  head                    ;; saves the head for a maneuvering
  stored_pitch            ;;
  x_last y_last z_last last_count  ;; save the last coordinates of the plane and how much time
is in the same point
  dist                    ;; distance flied count
  dist_H dist_V           ;; distance moved vertically and Horizontally
  time time_V time_H H_flag V_flag  ;; time the plane used to get to the target
  X_org                   ;; maneuvering flag
  Y_org                   ;; maneuvering flag
  desvioA desvioB desvioC  ;; maneuvering flags
]

UASs-own [
  destX destY destZ      ;; Destiny coordinates
  OrigX OrigY OrigZ      ;; Original Location
  colision                ;; Collision Flag
  angV                    ;; AngV is used when the diagonal running
  head                    ;; saves the head for a maneuvering
  stored_pitch            ;;
  x_last y_last z_last last_count  ;; save the last coordinates of the plane and how much time
is in the same point
  dist                    ;; distance flied count
  dist_H dist_V           ;; distance moved vertically and Horizontally
  time time_V time_H H_flag V_flag  ;; time the plane used to get to the target
  X_org                   ;; maneuvering flag
  Y_org                   ;; maneuvering flag
  desvioA desvioB desvioC  ;; maneuvering flags
]

patches-own [
  bloqueado ;; blocked flag
  quem      ;; who blocked the patch
  quando    ;; when the patch get blocked
  cb_state  ;; if the patch is a CB flag
]

```

Apêndice B.2 VARIÁVEIS GLOBAIS

```

.....
;; SETUP execution
;
.....

globals [
.....
; Simple Variables
;
.....

total_finish           ;; Número total de agentes que chegaram no destino
total_finish_UAS       ;; UAS finalizados
total_finish_VTOL      ;; VTOL finalizados
total_Colisions        ;; Número total de colisões
total_Conflits         ;; Número total de conflitos
total_time             ;; Time used to finish
total_time_VTOL        ;; Time used to finish VTOL
total_time_UAS         ;; Time used to finish UAS
total_time_Vertical    ;; Time used Vertical
total_time_Horizontal  ;; Time used Horizontal
total_time_Vertical_VTOL ;;
total_time_Horizontal_VTOL ;;
total_time_Vertical_UAS ;;
total_time_Horizontal_UAS ;;
dist_total            ;; Save the distance fly by the planes that finish the mission
dist_total_H          ;;
dist_total_V          ;;
TimeGeneration        ;; Time passed since last generation
LastTimeTotalPlanes   ;; Temp variable to save the total number of Planes (before creation
or destruction)
TxInput               ;; The number of planes create in this generation
TxOutput              ;; The number of planes destroyed in this generation
AEC                   ;; Air Space capacity (is TxInput/TxOutput)
Total_TxInput         ;;
Total_TxOutput        ;;
Destiny_type          ;; Says the destiny is -8 or 0
.....
; Parameters that need to be adjusted
; -- here the parameters collected in the the screen get converted
; to units to be used in the model
;
.....
Sep_horizontal_Unit
Sep_vertical_Unit
VelocidadeVertical_VTOL_Unit
VelocidadeVertical_UAS_Unit
VelocidadeHorizontal_VTOL_Unit
VelocidadeHorizontal_UAS_Unit
.....
; BATCH Variables
; -- this variables are used to save the results
; to be exported to the csv file
;
.....
fileName
lista_ticks
lista_total_finish
lista_total_Colisions
lista_total_Conflits
lista_total_Blocked

```

```
lista_AVG_Distance_Vertical  
lista_AVG_Distance_Horizontal  
lista_AVG_Time  
lista_AVG_Vertical  
lista_AVG_Horiz  
]
```

```
to clear  
  ;; Clear functions  
  reset-ticks  
  clear-turtles  
  clear-drawing  
  clear-all-plots  
  clear-output  
  clear-patches  
end
```

Apêndice B.3 *SETUP DE VARIÁVEIS*

```

.....
;; SETUP execution
;
.....

globals [
.....
; Simple Variables
;
.....

total_finish           ;; Número total de agentes que chegaram no destino
total_finish_UAS       ;; UAS finalizados
total_finish_VTOL      ;; VTOL finalizados
total_Colisions        ;; Número total de colisões
total_Conflits         ;; Número total de conflitos
total_time             ;; Time used to finish
total_time_VTOL        ;; Time used to finish VTOL
total_time_UAS         ;; Time used to finish UAS
total_time_Vertical    ;; Time used Vertical
total_time_Horizontal  ;; Time used Horizontal
total_time_Vertical_VTOL ;;
total_time_Horizontal_VTOL ;;
total_time_Vertical_UAS ;;
total_time_Horizontal_UAS ;;
dist_total             ;; Save the distance fly by the planes that finish the mission
dist_total_H           ;;
dist_total_V           ;;
TimeGeneration         ;; Time passed since last generation
LastTimeTotalPlanes    ;; Temp variable to save the total number of Planes (before creation
or destruction)
TxInput                ;; The number of planes create in this generation
TxOutput               ;; The number of planes destroyed in this generation
AEC                   ;; Air Space capacity (is TxInput/TxOutput)
Total_TxInput          ;;
Total_TxOutput         ;;
Destiny_type           ;; Says the destiny is -8 or 0
.....
; Parameters that need to be adjusted
; -- here the parameters collected in the the screen get converted
; to units to be used in the model
;
.....
Sep_horizontal_Unit
Sep_vertical_Unit
VelocidadeVertical_VTOL_Unit
VelocidadeVertical_UAS_Unit
VelocidadeHorizontal_VTOL_Unit
VelocidadeHorizontal_UAS_Unit
.....
; BATCH Variables
; -- this variables are used to save the results
; to be exported to the csv file
;
.....
fileName
lista_ticks
lista_total_finish
lista_total_Colisions
lista_total_Conflits
lista_total_Blocked

```

```
lista_AVG_Distance_Vertical  
lista_AVG_Distance_Horizontal  
lista_AVG_Time  
lista_AVG_Vertical  
lista_AVG_Horiz  
]
```

```
to clear  
  ;; Clear functions  
  reset-ticks  
  clear-turtles  
  clear-drawing  
  clear-all-plots  
  clear-output  
  clear-patches  
end
```


Apêndice B.4 CONFIGURAÇÕES DE CB

```

.....
;; CLIMA execution (CB) ;
.....

to set_cb [x y z] ;; set the CB in the patch xyz
ask patch x y z [
  set pcolor white
  trava -1 999999999999999
  set cb_state true
  set bloqueado true
]
end

to set_badClima ;; create one cloud in the CB
let zz random-zcor
if(zz < -7)[set zz -7]
let ry random-ycor
let rx random-xcor
set_cb rx ry zz
if(zz < 6) [set_cb rx ry zz + 1]
if(zz < 5) [set_cb rx ry zz + 2]
if(zz > -6) [set_cb rx ry zz - 1]
if(zz > -5) [set_cb rx ry zz - 2]
end

to set_badClima2 ;; create one cloud (type2) in the CB
let zz random-zcor
if(zz < -7)[set zz -7]
let ry random-ycor
let rx random-xcor
cloud_Var rx ry zz
let type_var (item (random 4) (list 1 2 3 4))
if(type_var = 1)[cloud_Var rx + 1 ry zz]
if(type_var = 2)[cloud_Var rx - 1 ry zz]
if(type_var = 3)[cloud_Var rx ry + 1 zz]
if(type_var = 4)[cloud_Var rx ry - 1 zz]
end

to cloud_Var [rx ry zz]
if(not passed_border rx ry zz) [
  set_cb rx ry zz
  if(zz < 6) [set_cb rx ry zz + 1]
  if(zz > -6) [set_cb rx ry zz - 1]
]
end

to set_clima ;; generate the CB
if(DÉBUG = "No"){
  let x 1
  while [x <= CB] [
    set_badClima
    set x (x + 1)
  ]
  set x 1
  while [x <= CB2] [
    set_badClima2

```

```
        set x (x + 1)
    ]
]
end
```

Apêndice B.5 BLOQUEIOS, DESVIOS, ETC.

```

.....
;; CLIMA execution (CB) ;
.....

to set_cb [x y z] ;; set the CB in the patch xyz
ask patch x y z [
  set pcolor white
  trava -1 999999999999999
  set cb_state true
  set bloqueado true
]
end

to set_badClima ;; create one cloud in the CB
let zz random-zcor
if(zz < -7)[set zz -7]
let ry random-ycor
let rx random-xcor
set_cb rx ry zz
if(zz < 6) [set_cb rx ry zz + 1]
if(zz < 5) [set_cb rx ry zz + 2]
if(zz > -6) [set_cb rx ry zz - 1]
if(zz > -5) [set_cb rx ry zz - 2]
end

to set_badClima2 ;; create one cloud (type2) in the CB
let zz random-zcor
if(zz < -7)[set zz -7]
let ry random-ycor
let rx random-xcor
cloud_Var rx ry zz
let type_var (item (random 4) (list 1 2 3 4))
if(type_var = 1)[cloud_Var rx + 1 ry zz]
if(type_var = 2)[cloud_Var rx - 1 ry zz]
if(type_var = 3)[cloud_Var rx ry + 1 zz]
if(type_var = 4)[cloud_Var rx ry - 1 zz]
end

to cloud_Var [rx ry zz]
if(not passed_border rx ry zz) [
  set_cb rx ry zz
  if(zz < 6) [set_cb rx ry zz + 1]
  if(zz > -6) [set_cb rx ry zz - 1]
]
end

to set_clima ;; generate the CB
if(DEBUG = "No")[
  let x 1
  while [x <= CB] [
    set_badClima
    set x (x + 1)
  ]
  set x 1
  while [x <= CB2] [
    set_badClima2

```

```
        set x (x + 1)
    ]
]
end
```

Apêndice B.6 EXECUÇÃO DO MODELO

```

.....
;; Move execution Control ;
.....

```

to loopMove [n]

```

.....
;; Move control in loop ;
;; Calls _loopMove_inner the number of times the plane should avance ;
;; if is up down or upfront is this same method ;
.....
let x 1
while [x <= n] [
  _loopMove_inner
  set x (x + 1)
]
validate_conflits_individual
end

```

to _loopMove_inner

```

.....
;; This function makes an individual move ;
;; verify the conflits/colisions after a moviment ;
;; have a colission avoid and unblock the cell ;
.....
move_away
..
;;
if(_move?)[
  fd 1
  set_dist
]
..
;;
move_back
..
;;
validate_colision_individual
destravaQuadrante
end

```

to set_dist

```

.....
;; This Function updates de traveled distance ;
.....
set dist (dist + 1)
ifelse(pitch = 0)[set dist_H (dist_H + 1)][set dist_V (dist_V + 1)]
end

```

to-report _move?

```

;; This function says if should move
ifelse(Detour_active) ;; If the flag is true will verify if will move
[report desvioB = false and patch_ok?]
[report true]
End

```

Apêndice B.7 GERAÇÃO E CONFIGURAÇÃO DE AERONAVES

```

.....
;; GENERATE execution ;
.....

to GenerateVTOL [N_max]
  create-VTOLs (random (N_max + 1)) [
    Generate red ;; VTOLs are red
  ]
end

to GenerateUAV [N_max]
  create-UASs (random (N_max + 1)) [
    Generate yellow ;; UAVs are yellow
  ]
end
.....

to Generate [colorValue]
  set_plane_params
  set color colorValue
  config_generate
  if(TraceCreation)[write who write xcor write ycor write destx write desty write "-"] ;; Create a Trace with
  info about the agents been created
end

to set_plane_params
  setxyz random-xcor random-ycor Destiny_type
  ..
  ;;
  set xcor ((floor xcor) - 0.5)
  set ycor ((floor ycor) - 0.5)
  ..
  ;;
  if(ycor < min-pycor) [set ycor min-pycor + 0.5]
  if(xcor < min-pxcor) [set xcor min-pxcor + 0.5]
  ..
  ;;
  set destX xcor
  set destY ycor
  set destZ Destiny_type
  ..
  ;;
  set angV 0
  ..
  ;;
  if(Scenario = "Parallel") [
    set destX random-xcor
  ]
  if(Scenario = "Perpendicular") [
    let c (random 2)
    if c = 0 [set destX random-xcor]
    if c = 1 [set destY random-ycor]
  ]
  ]
  ..
  ;;
  if(Scenario = "General") [
    let c (random 4)
    if c = 0 [set destX random-xcor]
    if c = 1 [set destY random-ycor]
    if c <= 2 [ ;; 50%
      let values (list 15 30 45 60 75 105 120 135 150 165 195 210 225 240 255 285 300 315 330 345)
      set angV (item (random 20) values)
      if(angV >= 0 and angV <= 90)[

```



```
]
if(heading >= 180)[
  ;; ft 1200 1600 2000 2400 2800 3200
  ;; 2 4 6 8 10 12 (-6 -4 -2 0 2 4)
  let itens (list)
  let size_itens 0
  if(Sep_vertical_Unit = 1) [ set itens (list -6 -4 -2 0 2 4) set size_itens 6]
  if(Sep_vertical_Unit = 2) [ set itens (list -6 -2 2) set size_itens 3]
  if(Sep_vertical_Unit = 3) [ set itens (list -6 2) set size_itens 2]
  if(Sep_vertical_Unit = 4)[ set itens (list -6) set size_itens 1]
  if(Sep_vertical_Unit = 5)[ set itens (list -6) set size_itens 1]
  set destZ (item (random size_itens) itens)
]
]
set head heading
set last_count 0
set x_last xcor
set y_last ycor
set z_last zcor
set dist 0
set desvioA false
set desvioB false
set desvioC false
set X_org 999
set Y_org 999
set OrigX xcor
set OrigY ycor
set OrigZ zcor
set H_flag false
set V_flag false
end
```


Apêndice B.8 MOVIMENTAÇÃO DE AERONAVES

```

.....
;; LOGIC execution ;
.....

to moveInd [VelocidadeVertical VelociadeHorizontal]
  _move_fd VelociadeHorizontal
  if(H_flag = False)[
    _move_down VelocidadeVertical
    _move_up VelocidadeVertical
  ]
  _move_update_time
  _move_checkpoint
end

to move_turtles
  ;; Call the move individual functions using the different velocity params
  ask turtles
  [
    if(breed = VTOLs)[moveInd VelocidadeVertical_VTOL_Unit VelocidadeHorizontal_VTOL_Unit]
    if(breed = UASs)[moveInd VelocidadeVertical_UAS_Unit VelocidadeHorizontal_UAS_Unit]
  ]
end

to verify_move
  ;; Verify if the agent have moved since last execution
  ask turtles [
    ifelse(xcor = x_last and ycor = y_last and zcor = z_last)[
      set last_count (last_count + 1)
    ]
    [ set x_last xcor
      set y_last ycor
      set z_last zcor
      set last_count 0 ]
  ]
end
.....

to _move_down [VelocidadeVertical]
  ;; descida
  if(_move_down_condition1 or _move_down_condition2) [
    if(destZ < zcor) [
      set V_flag True
      if(BlockCell_Active) [travaQuadrante_baixo (zcor + 7)]
      set pitch -90
      ..
      ..
      ifelse(destZ <= (zcor - VelocidadeVertical) )
        [loopMove VelocidadeVertical ]
        [loopMove (abs (destZ - zcor))]
      ..
      ..
      set pitch 0
      ..
      ..
      if(destZ = zcor) [set destZ Destiny_type] ;; If the destiny is different from the final objective set the
final objective
    ]
  ]
end

```

```

to-report _move_down_condition1
;; This condition says there is needed to get down
report (destZ < zcor and destZ != -8)
end

to-report _move_down_condition2
;; This condition says the plane is in the spot
report ((destX = xcor and destY = ycor) or
  ((abs(xcor - destX) <= 1) or (abs(ycor - destY) <= 1)) and (angV > 0))
end

to _move_up [VelocidadeVertical]
;; subida
if(destZ > Destiny_type)[
  set V_flag true
  if(BlockCell_Active and zcor = -8) [travaQuadrante_cima (destZ + 7)] ;; The lock only occur when zcor
  is -8
  set pitch 90
  ;;
  ifelse(destZ >= (zcor + VelocidadeVertical) )
    [loopMove VelocidadeVertical]
    [loopMove (abs (destZ - zcor))]
  ;;
  set pitch 0
  if(destZ = zcor) [set destZ Destiny_type]
]
end

to _move_fd [VelociadeHorizontal]
;; andando
if(destZ = Destiny_type)[ ;; Meta ir pro destino
  _move_fd_normal VelociadeHorizontal
  _move_fd_transversal VelociadeHorizontal
]
end

to _move_fd_transversal [VelociadeHorizontal]
if(angV != 0 and not _move_down_condition2) [
  set H_flag true
  ifelse(angV < 180)
  [
    ifelse(destX >= VelociadeHorizontal + xcor)
      [loopMove VelociadeHorizontal]
      [loopMove (abs (destX - xcor))]
  ]
  [
    ifelse(destX <= xcor - VelociadeHorizontal)
      [loopMove VelociadeHorizontal]
      [loopMove (abs (destX - xcor))]
  ]
]
ifelse(angV < 90 or angV > 270)
[
  ifelse(destY >= VelociadeHorizontal + ycor)
    [loopMove VelociadeHorizontal]
    [loopMove (abs (destY - ycor))]
  ]
[
  ifelse(destY <= ycor - VelociadeHorizontal)
    [loopMove VelociadeHorizontal]
  ]
]

```

```

    [loopMove (abs (destY - ycor))]
  ]
]
end

```

```

to _move_fd_normal [VelociadeHorizontal]
if(destX != xcor and angV = 0)[
  set H_flag true
  if(heading = 90)[
    ifelse(destX >= (xcor + VelociadeHorizontal) )
      [loopMove VelociadeHorizontal ]
      [loopMove abs (destX - xcor)]
  ]
  if(heading = 270)[
    ifelse(destX <= (xcor - VelociadeHorizontal))
      [loopMove VelociadeHorizontal ]
      [loopMove abs (destX - xcor)]
  ]
]
if(destY != ycor and angV = 0)[
  if(heading = 0) [
    ifelse(destY >= (ycor + VelociadeHorizontal) )
      [loopMove VelociadeHorizontal ]
      [loopMove abs (destY - Ycor)]
  ]
  if(heading = 180)[
    ifelse(destY <= (ycor - VelociadeHorizontal) )
      [loopMove VelociadeHorizontal ]
      [loopMove abs (destY - Ycor)]
  ]
]
]
end

```

```

to _move_update_time

```

```

.....
;; This function update the agent time vars ;
.....
set time time + 1
if(H_flag and V_flag)[write "OPS-001"]
if(H_flag = false and V_flag = false)[set time_H (time_H + 1) set H_flag false]
if(H_flag)[set time_H (time_H + 1) set H_flag false]
if(V_flag)[set time_V (time_V + 1) set V_flag false]
end

```

```

to _move_checkpoint

```

```

.....
;; This function verify if the agent is in the dest_location ;
.....
;; Get in the point normal
;;
if(xcor = destX and ycor = destY and zcor = destZ) [
  finish
]
;; In the ang moviment the cord are not not precise
if( ((abs(xcor - destX) <= 1) or (abs(ycor - destY) <= 1)) and (zcor = destZ) and (angV > 0)) [
  finish
]
]
end

```

```

to finish

```

```

;; This function Updates the finish the total_distance and destroy the agent

```

```

set total_time (total_time + time)
set total_time_Vertical (total_time_Vertical + time_V)
set total_time_Horizontal (total_time_Horizontal + time_H)
.....
if(breed = VTOLs)[set total_time_Vertical_VTOL (total_time_Vertical_VTOL + time_V)]
if(breed = VTOLs)[set total_time_Horizontal_VTOL (total_time_Horizontal_VTOL + time_H)]
.....
if(breed = UASs)[set total_time_Vertical_UAS (total_time_Vertical_UAS + time_V)]
if(breed = UASs)[set total_time_Horizontal_UAS (total_time_Horizontal_UAS + time_H)]
.....
if(breed = VTOLs)[set total_time_VTOL (total_time_VTOL + time)]
if(breed = UASs)[set total_time_UAS (total_time_UAS + time)]
.....
set total_finish (total_finish + 1)
if(breed = VTOLs)[set total_finish_VTOL (total_finish_VTOL + 1)]
if(breed = UASs)[set total_finish_UAS (total_finish_UAS + 1)]
.....
set dist_total (dist_total + dist)
set dist_total_H (dist_total_H + dist_H)
set dist_total_V (dist_total_V + dist_V)
.....
die
end

.....
;; Conflicts execution ;
.....

to validate_conflicts_individual
  _conflicts_horizontal
  _conflicts_up
  _conflicts_down
end

to validate_colision_individual
  let current_colisions 0
  if(zcor > -8 and count turtles-here > 1) [
    ask turtles-here [
      set colision True
      set current_colisions (current_colisions + 1)
    ]
  ]
  set total_Colisions ( total_Colisions + current_colisions)
  if(colision = True) [die]
end

.....

to _conflicts_horizontal
  .....
  ;; Frente ;
  .....
  _conflicts 0
end

to _conflicts_up
  .....
  ;; Em cima ;
  .....
  _conflicts 90
end

```

```
to _conflits_down
```

```
.....
;; Em baixo ;
.....
_conflits -90
end
```

```
to _conflits [p]
```

```
let curent_pitch pitch
set pitch p
let sep 0
ifelse(pitch = 0) [set sep Sep_horizontal_Unit][set sep Sep_vertical_Unit]
if(_conflits_inner sep)[set total_conflits (total_conflits + 1)] ;; Update the curent number of Conflicts
set pitch curent_pitch
end
```

```
to-report _conflits_inner [sep]
```

```
let x 1
while [x <= (sep - 1)] [
  let pos_x ([pxcor] of patch-ahead x)
  let pos_y ([pycor] of patch-ahead x)
  let pos_z ([pzcor] of patch-ahead x)
  if(not num_planes pos_x pos_y pos_z) [
    if(not passed_border pos_x pos_y pos_z)[
      report true
    ]
  ]
  set x (x + 1)
]
report false
end
```

```
.....
;; Bussiness Logic execution ;
.....
```

```
to GenerateLoad
```

```
if(DEBUG = "No")[
  ifelse(TimeGeneration > (TimeGenerationInterval - 1)) [
    ifelse(LimitPlanes)[
      ifelse((NumberMaxVTOL + NumberMaxVTOL2 + (count turtles)) >= LimitPlanesQnt)[
        let diff ((NumberMaxVTOL + NumberMaxVTOL2 + (count turtles)) - LimitPlanesQnt)
        let diff_1 ceiling (diff / 2)
        if(NumberMaxVTOL - diff_1 > 0) [GenerateVTOL NumberMaxVTOL - diff_1 - 1]
        if(NumberMaxVTOL2 - diff_1 > 0)[GenerateUAV NumberMaxVTOL2 - diff_1 - 1]
      ]
      [
        GenerateVTOL NumberMaxVTOL
        GenerateUAV NumberMaxVTOL2
      ]
    ]
    [
      GenerateVTOL NumberMaxVTOL
      GenerateUAV NumberMaxVTOL2
    ]
  ]
  set TimeGeneration 0
]
[
  set TimeGeneration (TimeGeneration + 1)
]
```

```

]
  set_txInput
]
end

```

```

to move
  move_turtles
  expire_lock
  verify_move
  set_txOutput
  set_aec
end

```

```

to set_txInput ;; Set the number of planes create this turn
  set TxInput ((count turtles) - LastTimeTotalPlanes)
  set LastTimeTotalPlanes (count turtles)
end

```

```

to set_txOutput ;; Set the number of planes Destroyed this turn
  set TxOutput (LastTimeTotalPlanes - (count turtles))
  set LastTimeTotalPlanes (count turtles)
end

```

```

to set_aec
  ;; Update the aec (airSpaceCapacity) in the turn
  if(DEBUG = "No")[
    set Total_TxInput (Total_TxInput + TxInput)
    set Total_TxOutput (Total_TxOutput + TxOutput)
    let turn_aec ( (TxInput + 1) / (TxOutput + 1) ) ;; TxOutput cannot be 0
    set AEC ((turn_aec * ticks) + AEC) / (ticks + 1)
  ]
end

```

```

to-report finish_simulation?
  ;; This function return true/false
  ;; if true is the desired condition to finish the simulation occurred
  if(Conclusion_type = "Time") [report (ticks >= MaxTime)] ;; Stop when the time passed
  if(Conclusion_type = "Accident") [report (total_Colisions >= 1)] ;; Stop when the first accident
  if(Conclusion_type = "Conflit") [report (total_Conflits >= 1)] ;; Stop when the first Conflict
  if(Conclusion_type = "Blocked") [ report (count turtles with [last_count > 10]) >= 1] ;; Stop when the first
  get blocked
end

```

```

.....
;; GO execution
.....

```

```

to Go
  GenerateLoad
  move
  ;;
  tick
  if(finish_simulation?) [ stop ] ;; if no agents left stop the loop
end

```

Apêndice B.9 EXECUÇÃO DO *BATCH*

```

.....
;; BATCH execution ;
.....

to batch_UpdateFile
  csv:to-file fileName (list
    lista_ticks
    lista_total_finish
    lista_total_Colisions
    lista_total_Conflits
    lista_total_Blocked
    lista_AVG_Distance_Vertical
    lista_AVG_Distance_Horizontal
    lista_AVG_Time
    lista_AVG_Vertical
    lista_AVG_Horiz
  )
end

to batch_clean_file
  set fileName (word "Results"
    "_" Scenario
    "_" NumberMaxVTOL "_" NumberMaxVTOL2 "_" TimeGenerationInterval "_" CB "CBs" "_"
    CB2 "CB2" "_" SizeWord
    "_" "VTOL" "_" VerticalSpeed_VTOL "FtMin" "_" SpeedHorizontal_VTOL "KT"
    "_" "VTOL2" "_" VerticalSpeed_VTOL2 "FtMin" "_" SpeedHorizontal_VTOL2 "KT"
    "_" "SepHorizontal" "_" Sep_horizontal "_" "SepVertical" "_" Sep_vertical
    ".csv")
  if file-exists? fileName [
    file-delete fileName
  ]
end

to batch_Run
  setup
  while [(not finish_simulation?) and (STOP_Var = false)]
  [
    go
  ]
end

to batch_Update_listas
  set lista_ticks (lput ticks lista_ticks)
  set lista_total_finish (lput total_finish lista_total_finish)
  set lista_total_Colisions (lput total_Colisions lista_total_Colisions)
  set lista_total_Conflits (lput total_Conflits lista_total_Conflits)
  set lista_total_Blocked (lput NumberConversion (count turtles with [last_count > 10])
  lista_total_Blocked)
  set lista_AVG_Distance_Vertical (lput NumberConversion ((dist_total_V / total_finish) * 200)
  lista_AVG_Distance_Vertical)
  set lista_AVG_Distance_Horizontal (lput NumberConversion ((dist_total_H / total_finish) * 0.5)
  lista_AVG_Distance_Horizontal)
  set lista_AVG_Time (lput NumberConversion (total_time / total_finish) lista_AVG_Time)
  set lista_AVG_Vertical (lput NumberConversion (total_time_Vertical / total_finish) lista_AVG_Vertical)
  set lista_AVG_Horiz (lput NumberConversion (total_time_Horizontal / total_finish) lista_AVG_Horiz)
end

```

