



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

## Utilização de Algoritmos Genéticos para Sequenciamento de Partidas em Aeroportos

Déborah Mendes Ferreira  
Lucas Pessoa Rosa

Brasília  
2013



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

## Utilização de Algoritmos Genéticos para Sequenciamento de Partidas em Aeroportos

Déborah Mendes Ferreira  
Lucas Pessoa Rosa

Monografia apresentada como requisito parcial  
para conclusão do Bacharelado em Ciência da Computação

Orientador  
Prof. Dr. Li Weigang

Brasília  
2013

Universidade de Brasília — UnB  
Instituto de Ciências Exatas  
Departamento de Ciência da Computação  
Bacharelado em Ciência da Computação

Coordenadora: Prof.<sup>a</sup> Dr.<sup>a</sup> Maristela Terto de Holanda

Banca examinadora composta por:

Prof. Dr. Li Weigang (Orientador) — CIC/UnB  
Prof. Dr. Flávio de Barros Vidal — CIC/UnB  
Prof.<sup>a</sup> Dr.<sup>a</sup> Maristela Terto de Holanda — CIC/UnB

#### **CIP — Catalogação Internacional na Publicação**

Ferreira, Déborah Mendes.

Utilização de Algoritmos Genéticos para Sequenciamento de Partidas em Aeroportos / Déborah Mendes Ferreira, Lucas Pessoa Rosa. Brasília : UnB, 2013.

155 p. : il. ; 29,5 cm.

Monografia (Graduação) — Universidade de Brasília, Brasília, 2013.

1. algoritmos genéticos, 2. alocação de *slots*, 3. gerenciamento de partidas, 4. CDM

CDU 004.4

Endereço: Universidade de Brasília  
Campus Universitário Darcy Ribeiro — Asa Norte  
CEP 70910-900  
Brasília-DF — Brasil



Universidade de Brasília

Instituto de Ciências Exatas  
Departamento de Ciência da Computação

## Utilização de Algoritmos Genéticos para Sequenciamento de Partidas em Aeroportos

Déborah Mendes Ferreira

Lucas Pessoa Rosa

Monografia apresentada como requisito parcial  
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Li Weigang (Orientador)

CIC/UnB

Prof. Dr. Flávio de Barros Vidal    Prof.<sup>a</sup> Dr.<sup>a</sup> Maristela Terto de Holanda

CIC/UnB

CIC/UnB

Prof.<sup>a</sup> Dr.<sup>a</sup> Maristela Terto de Holanda

Coordenadora do Bacharelado em Ciência da Computação

Brasília, 19 de dezembro de 2013

# Dedicatória

Dedicamos aos professores que nos inspiraram ao longo de todos estes semestres.

# Agradecimentos

Agradecemos à nossas famílias por todo apoio e carinho.

Agradecemos ao professor Li Weigang pela orientação e por todas as oportunidades.

Agradecemos à FINEP e à Atech pelo suporte às nossas pesquisas.

Agradecemos ao senhor Antenor do Centro de Operações do Aeroporto Internacional de Brasília por nos dar a oportunidade de conhecer todo o funcionamento e estrutura do aeroporto.

# Resumo

A Tomada de Decisão Colaborativa é um conjunto de artefatos operacionais onde todos os agentes envolvidos no gerenciamento do fluxo trafego aéreo(ATFM) devem participar ativamente. No Brasil todo o gerenciamento é feito de forma manual pelos controladores de voo, que se utilizam da experiência para organizar e sequenciar os voos que irão decolar.

A presente proposta têm o intuito de tratar o problema do gerenciamento de partidas em aeroportos e se baseia em um gerenciamento colaborativo e no sequenciamento, distribuição e alocação automática de *slots*.

O objetivo deste trabalho é propor uma solução utilizando algoritmos genéticos que empregando conceitos de Tomada de Decisão Colaborativa, e utilizando os conceitos dos algoritmos, *Ration by Schedule* e *Compression*, seja capaz de diminuir o atraso em aeroportos.

Como solução foi desenvolvido um protótipo em linguagem C onde simulações foram realizadas utilizando planos de voos reais válidos para o dia 20 de novembro de 2013. O protótipo gerou resultados relevantes. O sequenciamento permitiu reduzir os atrasos em até 34,85% em horário de pico, comprovando a efetividade do modelo.

**Palavras-chave:** algoritmos genéticos, alocação de *slots*, gerenciamento de partidas, CDM

# Abstract

The Collaborative Decision Making(CDM) is an operational artifact collection where all the stakeholders involved in the Air Traffic Flow Management (ATFM) should actively participate. In Brazil all the flow management is done manually by the flights controller's, who use their experience to organize and sequence the flight departures.

The current research proposal was created to address the departure sequencing problem in airports and is based on an approach to collaborative departure sequencing, distribution and automatic slot allocation.

The goal of this work is to propose a solution using Genetic Algorithms that implements CDM concepts, and use Ration By Schedule and Compression algorithms to reduce the delay in flights.

A solution was developed as a prototype in C language where simulations were performed using valid real plans valid for flights on November 20, 2013. The prototype has generated significant results. The sequencing allowed for the reduction of delays in up to 34.85% in rush hour, proving the effectiveness of the model.

**Keywords:** genetics algorithms, slot allocation, departure management, CDM



# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação . . . . .	2
1.2	Objetivos . . . . .	2
1.3	Metodologia . . . . .	2
1.4	Organização do Trabalho . . . . .	3
<b>2</b>	<b>Algoritmos Genéticos</b>	<b>4</b>
2.1	Conceitos . . . . .	4
2.2	Histórico . . . . .	5
2.3	Funcionamento dos AG . . . . .	5
2.3.1	Inicialização . . . . .	6
2.3.2	Avaliação . . . . .	6
2.3.3	Seleção . . . . .	6
2.3.4	<i>Crossover</i> . . . . .	8
2.3.5	Mutação . . . . .	8
2.3.6	Substituição . . . . .	9
2.3.7	Término da Execução . . . . .	10
2.4	Paralelismo em Algoritmos Genéticos . . . . .	11
2.4.1	Populações Globais com Paralelismo . . . . .	11
2.4.2	Modelo de ilhas . . . . .	11
2.5	Aplicações dos Algoritmos Genéticos . . . . .	12
2.5.1	Hibridização . . . . .	12
2.5.2	Teoria dos Jogos com AG . . . . .	13
<b>3</b>	<b>Conceitos de ATFM</b>	<b>15</b>
3.1	Conceitos Gerais . . . . .	15
3.1.1	Serviços de Tráfego Aéreo - ATS . . . . .	15
3.2	Gerenciamento de Tráfego Aéreo . . . . .	16
3.3	Gerenciamento do Fluxo de Tráfego Aéreo . . . . .	17
3.4	Fases do Voo . . . . .	17
3.5	Restrições operacionais . . . . .	18
3.5.1	Controle de saturação . . . . .	18
3.5.2	Procedimentos pré-decolagem . . . . .	19
3.5.3	Separação entre aeronaves . . . . .	20
3.6	O Tráfego Aéreo no Brasil . . . . .	20
3.7	CSM/ATM . . . . .	21

<b>4</b>	<b>Collaborative Decision Making - CDM</b>	<b>23</b>
4.1	Funcionamento . . . . .	23
4.2	Alocação de Slots . . . . .	25
4.2.1	Histórico . . . . .	25
4.2.2	Alocação de <i>Slots</i> no Brasil . . . . .	26
4.3	RBS - <i>Ration-by-Schedule</i> . . . . .	27
4.4	Compression . . . . .	28
<b>5</b>	<b>Estado da Arte</b>	<b>30</b>
5.1	Alocação de <i>Slots</i> utilizando Teoria dos Jogos . . . . .	30
5.1.1	Teoria dos Jogos no Sistema CoDMAN . . . . .	31
5.1.2	Modelagem do Cenário . . . . .	31
5.1.3	Modelagem das Aeronaves . . . . .	31
5.1.4	Negociação de <i>Slots</i> . . . . .	32
5.1.5	Avaliação dos Resultados . . . . .	33
5.2	Gerenciamento de decolagem: Economizando o tempo de táxi, queima de combustível e emissões . . . . .	33
5.2.1	Implementação e Objetivos . . . . .	34
5.2.2	Benefícios da ferramenta DMAN . . . . .	34
5.3	Realocação de recursos dos aeroportos em condições meteorológicas adversas. . . . .	35
5.4	SIVOR - <i>Simulateur de VOI en Route</i> . . . . .	36
<b>6</b>	<b>Sequenciamento de Partidas de Aeronaves</b>	<b>38</b>
6.1	Problema . . . . .	38
6.2	Aeroporto Internacional de Brasília . . . . .	39
6.3	Sequenciamento de Partidas no Aeroporto de Brasília . . . . .	39
6.4	Coleta de Dados . . . . .	41
6.5	Atrasos e Cancelamentos . . . . .	41
<b>7</b>	<b>Modelagem e Implementação</b>	<b>43</b>
7.1	Alocando <i>slots</i> . . . . .	43
7.2	Modelagem do Algoritmo Genético . . . . .	44
7.2.1	Representação genética . . . . .	44
7.2.2	Função <i>fitness</i> . . . . .	46
7.3	Restrições na Alocação de <i>slots</i> . . . . .	47
7.4	Implementação da proposta . . . . .	48
<b>8</b>	<b>Estudo de Caso</b>	<b>53</b>
8.1	Ambiente de Teste . . . . .	53
8.2	Planejamento de Casos de Testes . . . . .	53
8.3	Cenário 1: Horário de pico . . . . .	54
8.4	Cenário 2: Horário normal . . . . .	54
8.5	Resultados Obtidos a partir da Proposta . . . . .	54
8.5.1	Tamanho da População . . . . .	54
8.5.2	Número de gerações . . . . .	55
8.5.3	Resultados: Cenário 1 . . . . .	55
8.5.4	Resultados: Cenário 2 . . . . .	56

8.6	Comparação entre os Dados Reais e os Dados Obtidos. . . . .	57
8.7	Resultado da Alocação de <i>Slots</i> . . . . .	57
8.8	Comparação com CoDMAN . . . . .	58
<b>9</b>	<b>Considerações Finais</b>	<b>61</b>
9.1	Avaliação dos Resultados . . . . .	61
9.2	Trabalhos Futuros . . . . .	62
	<b>Referências</b>	<b>63</b>

# Abreviaturas

**AA** Área de Absorção.

**ACC** Centro de Controle de Área (*Area Control Center*).

**AG** Algoritmos Genéticos.

**ANAC** Agência Nacional de Aviação Civil.

**APP** Controle de Aproximação (*Approximation*).

**ATC** Controle de Tráfego Aéreo (*Air Traffic Control*).

**ATFM** Gerenciamento de Fluxo de Tráfego Aéreo (*Air Traffic Flow Management*).

**ATM** Gerenciamento de Tráfego Aéreo (*Air Traffic Management*).

**ATS** Serviço de Tráfego Aéreo (*Air Traffic Service*).

**CDM** Tomada Colaborativa de Decisão (*Collaborative Decision Making*).

**CGNA** Centro de Gerenciamento da Navegação Aérea.

**CoDMAN** Gerenciamento Colaborativo de Partidas (*Collaborative Departure Management*).

**CTOT** Horário Calculado de Decolagem (*Calculated Takeoff Time*).

**DECEA** Departamento de Controle do Espaço Aéreo.

**DMAN** Gerenciamento de Partidas (*Departure Management*).

**EIBT** Horário Estimado de Parada Total (*Estimated In-block Time*).

**ETOT** Horário Estimado de Decolagem (*Estimated Takeoff Time*).

**Eurocontrol** Organização Europeia para a Segurança da Navegação Aérea (*European Organization for the Safety of Air Navigation*).

**EXIT** Horário Estimado de Taxi-In (*Estimated taXi-In Time*).

**EXOT** Horário Estimado de Taxi-Out (*Estimated taXi-Out Time*).

**FAA** Administração Federal de Aviação (*Federal Aviation Administration*).

**FIFO** Primeiro a chegar, primeiro a sair (*First in, first out*).

**GDP** Programa de Atraso em Solo(*Ground Delay Program*).

**Hotran** Horário de Transporte.

**IFR** Regras de Voo Instrumentado (*Instrumented Flight Rules*).

**JFK** Aeroporto Internacional John F. Kennedy(*John F. Kennedy International Airport*).

**NASA** Administração Nacional da Aeronáutica e do Espaço(*National Aeronautics and Space Administration*).

**PHL** Aeroporto Internacional da Philadelphia(*Philadelphia International Airport*).

**RBS** *Ration by Schedule*.

**SIVOR** Simulador de voo em rota(*SImulateur de VOI en Route*).

**TDD** Meta de Horário para Decolagem (Target Time for Departure).

**TMA** Área Terminal de Controle.

**TOBT** Meta de Tempo para Início de Movimentação (*Target Off-Block Time*).

**TransLab** Laboratório de Modelo Computacional para Transporte Aéreo.

**TWR** Área de Controle de Torre (*Tower*).

**VFR** Regras de Voo Visual (*Visual Flight Rules*).

# Lista de Figuras

2.1	Método da Roleta[15]. . . . .	7
2.2	Seleção por torneio. . . . .	8
2.3	Tipos de <i>crossover</i> . . . . .	9
2.4	<i>Crossover</i> uniforme baseado em ordem. . . . .	10
2.5	Um exemplo de mutação. . . . .	10
2.6	Modelo de ilhas[56]. . . . .	12
2.7	Matriz de <i>payoff</i> do dilema do prisioneiro. A letra C representa cooperação e a letra T representa traição. . . . .	13
3.1	Processos de decolagem [25]. . . . .	18
4.1	Compartilhamento de informações nos aeroportos da Europa que utilizam o CDM[25]. . . . .	24
4.2	Participação no mercado das empresas aéreas em 2011[4]. . . . .	27
4.3	Dados dos <i>slots</i> de cada empresa aérea[46]. . . . .	28
4.4	Exemplo de alocação utilizando <i>Compression</i> . . . . .	29
5.1	Realocação de <i>slots</i> utilizando <i>Top Trading Cycles</i> [7]. . . . .	36
6.1	Diagrama do Aeroporto Internacional de Brasília [4]. . . . .	40
7.1	Alocação dos <i>slots</i> . . . . .	44
7.2	Alocação dos <i>slots</i> em horários próximos. . . . .	44
7.3	Estrutura dos genes e do cromossomo. . . . .	45
7.4	O voo da companhia aérea que liberou o <i>slot</i> não pode utilizar este <i>slot</i> , então outro voo foi realocado para melhorar o <i>slot</i> de outro voo da GOL. . . . .	47
7.5	O voo da companhia aérea que liberou o <i>slot</i> pode utilizar o <i>slot</i> liberado. . . . .	47
7.6	Nenhum voo pode utilizar o <i>slot</i> liberado. . . . .	48
7.7	Diagrama representando a implementação. . . . .	48
7.8	Fluxograma da implementação proposta. . . . .	50
7.9	Fluxograma da alocação de slots. . . . .	51
7.10	Fluxograma do algoritmo genético. . . . .	52
8.1	Gerações para convergência do Algoritmo Genético. . . . .	55
8.2	Tempo de convergência do Algoritmo Genético. . . . .	56

# Lista de Tabelas

5.1	Comparação do desempenho dos resultados obtidos para a diminuição do custo total do cenário. . . . .	33
5.2	Redução no tempo de táxi na área de movimentação do Aeroporto JFK. . .	35
5.3	Redução no tempo de atraso do Aeroporto JFK. . . . .	35
6.1	Dados do Aeroporto Internacional de Brasília em outubro de 2013 [30]. . .	39
6.2	Dados de partida dos voos obtidos do site da Infraero [29]. . . . .	41
7.1	Códigos para algumas das empresas aérea. . . . .	45
8.1	Ambiente de testes . . . . .	53
8.2	Resultados obtidos com o modelo proposto no Cenário 1. . . . .	56
8.3	Resultados obtidos com o modelo proposto, no Cenário 2. . . . .	56
8.4	Comparação entre os dados reais e os dados obtidos. . . . .	57
8.5	Voos no horário de pico. . . . .	59
8.6	Voos no horário de pico com aplicação da proposta. . . . .	60
9.1	Comparação entre os dados reais e os melhores dados obtidos. . . . .	62

# Capítulo 1

## Introdução

A taxa de voos atrasados em aeroportos de grande porte no Brasil pode chegar a uma média de 6% por dia, chegando ao triplo desse valor em dias de condições meteorológicas adversas[39]. Quando ocorrem atrasos as aeronaves passam parte do tempo esperando autorização para decolagem e outra parte em uma fila à espera da liberação da pista de decolagem.

A espera na fila para uso da pista de decolagem é realizada com os motores ligados, o que gera um dispêndio de combustível e emissão de poluentes no meio ambiente, além de prejuízos às companhias aéreas que perdem passageiros, aos aeroportos que perdem voos e aos passageiros que perdem tempo e dinheiro. De acordo com a Administração Federal de Aviação (*Federal Aviation Administration*) (FAA) [26], os atrasos nos voos nos Estados Unidos geram um prejuízo aproximado de 22 bilhões de dólares anualmente para as companhias aéreas.

O sequenciamento de aeronaves em partida é o processo de pré-decolagem, em que os voos são distribuídos em períodos de tempo no qual os mesmos devem estar dentro no processo de decolagem. Estes períodos de tempo são chamados de *slots*, que geralmente tem uma duração de 1 a 5 minutos, dependendo da situação da pista. Quando um voo atrasa, seu *slot* deve ser realocado para que a pista não fique vazia e este voo atrasado precisará de um novo *slot* para que possa decolar em um outro horário.

Para que seja possível realizar estas modificações, um sistema de sequenciamento de partidas é necessário. O sequenciamento ineficiente de aeronaves para partida também pode ser responsável por aumentar o atraso dos voos e ainda colaborar com o aumento das filas físicas antes das decolagens.

O Aeroporto Internacional de Brasília não utiliza *slots*, os horários de partida são baseados no Hotran [1], que é um horário pré-definido que a aeronave tem autorização para decolar, o uso da pista para decolagem é feito, geralmente, de acordo com um esquema *first-come, first-served*. Os *slots* são utilizados em todo o tráfego aéreo europeu e, em geral, se mostraram benéficos[13]. Os *slots* fazem com que os voos se distribuam igualmente ao longo do dia, evitando que ocorram horários de muito congestionamento e atrasos.

A atividade de sequenciamento no Brasil é realizada de forma empírica pelos controladores de voo, que devem negociar com as empresas de transporte aéreo o posicionamento das aeronaves sem um critério claro, o que pode acarretar em mais problemas para o controlador, que já precisa lidar com outras situações complexas em sua profissão.



## 1.1 Motivação

O gargalo de fluxo do transporte aéreo não está mais associado ao tráfego aéreo em si, mas na capacidade dos aeroportos, áreas com o espaço normalmente limitado, de gerir com fluidez o tráfego em solo que tem aumentado ao longo dos anos.

No ano de 2014, o Brasil passará por um período crítico na aviação brasileira devido à Copa do Mundo de 2014. A previsão é que os aeroportos superem em 50% suas capacidades máximas, gerando uma grande superlotação e longos atrasos[9]. Caso as obras dos aeroportos não sejam concluídas, a previsão é que a situação seja ainda pior, por exemplo, o Aeroporto de Belo Horizonte pode chegar à 260% de sua capacidade. É fundamental criar medidas que possam diminuir estes atrasos.

O grupo de estudos do Laboratório de Modelo Computacional para Transporte Aéreo (TransLab) tem se dedicado na criação e estudo de ferramentas que implementem à Tomada Colaborativa de Decisão (*Collaborative Decision Making*) (CDM) para auxiliar os controladores de voo em solo, tanto na definição de *taxiways* [39], como no sequenciamento de aeronaves para partida [53].

## 1.2 Objetivos

Este trabalho tem por objetivo apresentar uma proposta de solução para o gerenciamento eficiente de partidas de aeronaves nos aeroportos, através de um modelo de alocação e sequenciamento de aeronaves.

Os objetivos específicos são:

- Simular cenários distintos de condições dos aeroportos, tentando cobrir as situações especiais e verificar a aplicabilidade do modelo a essas situações;
- Desenvolvimento de uma solução para o sequenciamento de decolagens utilizando algoritmos genéticos e avaliados de acordo com os princípios dos algoritmos *Ration by Schedule* (RBS) e *Compression*, utilizando alocação de *slots*.

## 1.3 Metodologia

O trabalho foi concebido através da seguinte metodologia de pesquisa:

- Levantamento de requisitos: os requisitos são levantados junto aos especialistas e usuários que são os controladores, são levantados também os recursos necessários para o desenvolvimento da solução.
- Levantamento de informações: esses levantamentos permitirão uma maior compreensão dos procedimentos utilizados pelos controladores na prática.
- Levantamento do estado da arte: é realizado um estudo de trabalhos que direta ou indiretamente estão relacionados a esta pesquisa, permitindo um melhor conhecimento dos problemas encontrados, as soluções propostas e observar o que não foi resolvido em outras pesquisas.

- Definição da arquitetura: nessa etapa será definida como se organizará a solução, como os dados serão estruturados e em que linguagem será desenvolvida.
- Implementação: nessa etapa serão aplicados os conceitos levantados nas etapas anteriores em busca de uma solução para o modelo.
- Simulação: os casos de testes e os cenários que serão simulados pela aplicação são definidos e executados nessa fase.
- Análise dos resultados: os dados obtidos nas simulações são analisados e a eficiência do modelo é calculada.

## 1.4 Organização do Trabalho

O Capítulo 2 trata dos conceitos de algoritmos genéticos que são utilizados para a realização deste trabalho, apresentando uma breve introdução, seu funcionamento e aplicações.

O Capítulo 3 trata sobre os conceitos de Gerenciamento de Fluxo de Tráfego Aéreo (*Air Traffic Flow Management*) (ATFM), necessário para um melhor entendimento dos atores, termos e estruturas utilizadas no contexto do gerenciamento de tráfego aéreo.

O Capítulo 4 discorre sobre a Tomada de Decisão Colaborativa, iniciativa que visa uma participação de atores, interessados na questão foco, na tomada de decisão. Também neste capítulo são apresentados os conceitos de RBS e *Compression*.

O Capítulo 5 apresenta projetos realizados na área fim deste trabalho, demonstrando a viabilidade das ideias apresentadas por este trabalho.

O Capítulo 6 descreve em detalhes como ocorre o sequenciamento de partidas em aeroportos e como ocorre no Aeroporto Internacional de Brasília, que será utilizado como estudo de caso.

O Capítulo 7 apresenta o modelo proposto, o ambiente de desenvolvimento e a implementação realizada para solucionar o problema de sequenciamento de partida. No Capítulo 8 são tratados os resultados e a análise dos mesmos.

O Capítulo 9 traz a conclusão do trabalho apresentando os resultados frente aos objetivos propostos e apresenta propostas de trabalhos futuros que foram observados durante o desenvolvimento deste.

# Capítulo 2

## Algoritmos Genéticos

Os Algoritmos Genéticos (AG) são métodos de busca baseados nos princípios da seleção natural e na genética [56]. Os AG codificam as variáveis de decisão de problemas de busca como *strings* finitas formadas por alfabetos de uma certa cardinalidade. As *strings* que formam as possíveis soluções são chamadas de **cromossomos**, os alfabetos utilizados são chamados de **genes** e o valor dos genes são chamados de **alelos**. Por exemplo, para resolver o Problema do Caixeiro Viajante[43] utilizando AG, os cromossomos representariam as rotas possíveis e os genes representariam as cidades.

Os conceitos iniciais de algoritmos genéticos são apresentados na Seção 2.1 e um breve histórico é apresentado na Seção 2.2. O funcionamento e as etapas dos AG são detalhados na Seção 2.3, na Seção 2.4 são apresentadas formas de se paralelizar o algoritmo para diminuir seu tempo de execução. Por fim, algumas aplicações são apresentadas na Seção 2.4.

### 2.1 Conceitos

Os algoritmos genéticos possuem este nome pois fazem uma analogia com a biologia, uma **população** de *strings* é utilizada e essa população é recombinaada através do **cross-over** e da **mutação** [56]. A busca é guiada pelos resultados da avaliação da função objetiva  $f$  para cada *string* na população. Baseando-se nesta avaliação, *strings* que possuem maior **fitness** possuem maior probabilidade de se reproduzirem.

Algumas vantagens dos algoritmos genéticos são:

1. Otimização com variáveis contínuas ou discretas;
2. Não é necessária informação sobre a derivada;
3. Lida com um grande número de variáveis;
4. Pode ser implementado utilizando programação paralela;
5. Fornece uma lista de resultados ótimos, não apenas uma única solução;
6. Pode ser utilizado com dados gerados, dados experimentais ou funções analíticas;
7. Busca a solução simultaneamente em vários locais.

Apesar das suas vantagens, nem sempre os AG são adequados para um determinado problema. Existem algumas funções que conseguem “esconder” a solução ótima (aquela solução que mais se aproxima do objetivo esperado) dos algoritmos genéticos, estas funções são chamadas de funções enganosas, elas fazem com que os AG busquem uma falsa solução ótima, nessa situação a solução ótima só pode ser alcançada por pura sorte[47].

Uma função analítica bem comportada (contínua e finita) de poucas variáveis pode ser facilmente calculada utilizando técnicas do Cálculo, que apresentarão um desempenho superior aos AG. A grande população de soluções geradas pelos AG tem como consequência uma queda no desempenho, mas que pode ser parcialmente compensado com a utilização do paralelismo.

Algumas desvantagens dos AG são:

1. Alguns problemas de otimização não podem ser resolvidos por algoritmos genéticos.
2. Não há maneira de ter certeza absoluta que o algoritmo irá encontrar o ótimo global.
3. Não é possível garantir um tempo de resposta otimizada constante.

## 2.2 Histórico

O termo algoritmo genético foi primeiramente usado por John Holland[32], ele escreveu o livro *Adaptation in Natural and Artificial Systems* em 1975 e sua aplicação vai além do original algoritmo genético. A influência de Holland neste assunto é muito importante, mas diversos outros cientistas também desenvolveram ideias similares.

Em 1960, na Alemanha, Ingo Rechenberg[31] e Hans-Paul Schwefel[44] desenvolveram o *Evolutionstrategie* (traduzido como estratégia de evolução). Na mesma época, nos Estados Unidos, Bremermann e Fogel implementaram uma ideia que eles chamaram de programação evolucionária. O tema comum de todas estas ideias é a utilização da mutação e da seleção. Os resultados pareciam promissores, mas só conseguiram sucesso a partir da década de 80. Essas técnicas precisavam de um grande poder computacional, que não estava disponível na época[15].

A otimização não era o foco principal do trabalho do Holland em sistemas adaptativos, mesmo assim a maioria dos trabalhos utilizam AG com esse propósito. O trabalho de um de seus alunos, Ken DeJong[35], mostrou a capacidade de otimização dos algoritmos genéticos. Finalmente, os algoritmos genéticos foram popularizados por um outro aluno de Holland, David Goldberg[20], que conseguiu resolver o difícil problema de controle de transmissão de um oleoduto de gás, utilizando essa técnica. Desde então, muitas versões da programação evolucionária têm sido experimentadas com variáveis graus de sucesso.

## 2.3 Funcionamento dos AG

Os passos para se evoluir a solução para o problema de busca são os seguintes: inicialização, avaliação, seleção, *crossover*, mutação e substituição. Cada uma destas etapas são descritas nesta seção.

### 2.3.1 Inicialização

Após definir o problema em forma de *strings* de cromossomos, pode-se iniciar a execução do algoritmo. Para gerar a população inicial, é necessário considerar seu tamanho e como os seus indivíduos serão escolhidos. É necessário considerar um equilíbrio entre eficiência e efetividade. Uma população muito pequena não teria espaço suficiente para buscar uma solução e uma população muito grande traria problemas para o desempenho. Em termos de eficiência, a melhor opção é se escolher o menor tamanho de população possível para que uma busca significativa ocorra.

Para a escolha da população inicial, o mais comum é que a população seja gerada de forma aleatória[15], mas também é possível “semear” uma solução conhecida que seja uma boa solução para o algoritmo, isto pode ajudar os AG a encontrarem soluções melhores de forma mais rápida[14].

### 2.3.2 Avaliação

Com a população criada ou após uma nova geração ser originada, é necessário avaliar o valor de *fitness* dos candidatos à solução. O valor de *fitness* é um valor numérico que expressa o desempenho de um indivíduo em relação ao ótimo procurado, permitindo que diferentes indivíduos sejam comparados.

O nível de sucesso da aplicação dos AG depende criticamente da definição da função *fitness*, esta função não deve variar nem rápido demais, nem devagar demais. A função deve garantir que os indivíduos possam ser diferenciados de acordo com sua adequação para resolver o problema de otimização[6].

Para se garantir que uma expressão condicional “ $x=y$ ” seja sempre verdadeira, pode-se definir uma função *fitness*  $f = |x - y|$ . A busca de soluções teria como objetivo encontrar uma população com o menor valor de *fitness* possível. Dependendo de como a função for definida, pode ser que a melhor população seja aquela com maior ou menor valor de *fitness*.

Na Biologia, o conceito de *fitness* está relacionado com a capacidade de organismos sobreviverem e se reproduzirem no ambiente em que se encontram. Os indivíduos com maior *fitness* conseguem contribuir com seus genes para a próxima geração[3]. Ou seja, as características desse indivíduo vão ter uma maior probabilidade de se perpetuar. Nos algoritmos genéticos busca-se perpetuar as soluções que apresentam bons resultados.

### 2.3.3 Seleção

Na etapa de seleção, os indivíduos com um melhor valor de *fitness* serão escolhidos com uma maior chance de se reproduzirem, garantindo assim a sobrevivência do mais adaptado (“*survival of the fittest*”). Nesta Seção são apresentadas duas classes de seleção.

#### Seleção Proporcional ao *Fitness*

O método mais comum dentro desta classe é o método da roleta, neste método cada indivíduo recebe um espaço na roleta proporcional ao seu valor de *fitness*. Ou seja, as melhores soluções terão um espaço maior na roleta do que as soluções com menor valor

de *fitness*. A Figura 2.1 ilustra este método. Os passos para se implementar o método da roleta são[36]:

1. Avalie o *fitness*,  $f_i$ , de cada  $i$  indivíduo da população.
2. Calcule a probabilidade(tamanho do espaço na roleta),  $p_i$ , de se selecionar cada indivíduo da população:  $p_i = \frac{f_i}{\sum_{j=1}^n f_j}$ , onde  $n$  é o tamanho da população.
3. Calcule a propabilidade cumulativa,  $q_i$ , para cada indivíduo:  $q_i = \sum_{j=1}^i f_j$ .
4. Gere um número uniforme aleatório,  $r \in (0, 1]$
5. Se  $r < q_1$ , selecione o primeiro cromossomo,  $x_1$ , senão escolha o indivíduo  $x_i$  tal que  $q_{i-1} < r \leq q_i$
6. Repita os passos 4 e 5  $n$  vezes para criar  $n$  candidatos na área de reprodução.

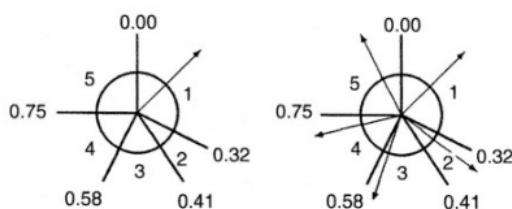


Figura 2.1: Método da Roleta[15].

A Figura 2.1 mostra um exemplo do Método da Roleta, nestas roletas existem 5 *strings*, a probabilidade de se escolher um determinado indivíduo é proporcional ao tamanho do seu setor na roleta. Os valores no exterior da roleta são as probabilidades cumulativas, elas serão “sorteadas” com a geração de um número aleatório.

### Seleção Ordinal

Esta classe inclui os métodos de seleção por torneio e a seleção truncada. Na seleção por torneio  $s$  cromossomos são escolhidos de forma aleatória e inseridos em um torneio, um contra o outro. O indivíduo com maior valor de *fitness* do grupo de  $k$  cromossomos ganha o torneio e é selecionado como pai. O valor mais usado para  $s$  é 2. Desta forma, são necessários  $n$  torneios para se escolher  $n$  indivíduos. Na seleção truncada, os primeiros  $(1/s)$ -ésimos indivíduos ganham  $s$  cópias cada na área de reprodução.

A Figura 2.2 exhibe um exemplo de seleção por torneio, na primeira rodada os indivíduos  $x$ ,  $w$ ,  $y$  e  $z$  competem para entrar na área de reprodução, os indivíduos  $x$  e  $y$  vencem a rodada e disputam a próxima, ambos ganham uma cópia na área de reprodução na rodada 2. O  $y$  ganha a rodada 2 e consegue outra cópia na área de reprodução durante a rodada 3, aumentando sua chance de propagar seus genes.

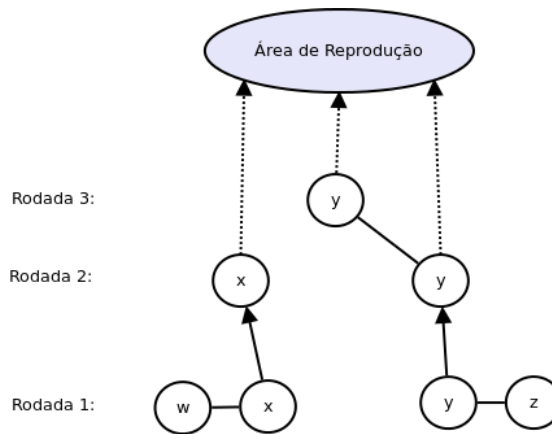


Figura 2.2: Seleção por torneio.

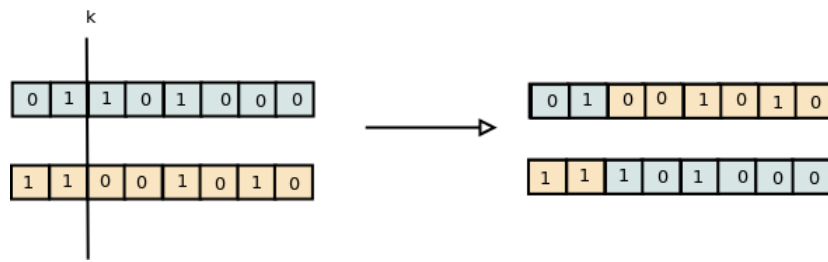
### 2.3.4 Crossover

Após a etapa de seleção, os indivíduos que estão na área de reprodução sofrem *crossover* para criar novas gerações, que espera-se que sejam melhores que a anterior. No *crossover* os genes de um genitor são substituídos pelos genes de um outro correspondente. Os tipos mais comuns de crossover são:

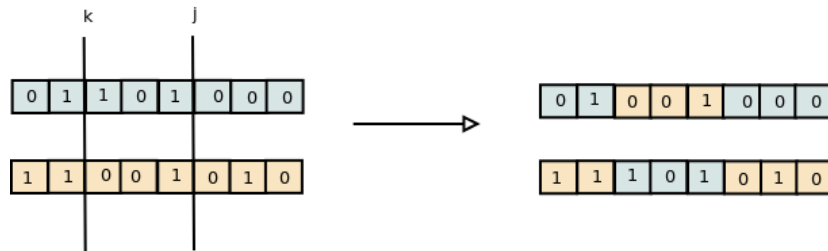
- *Crossover* em um ponto (Figura 2.3(a)): Neste *crossover* se escolhe um ponto aleatório,  $k$ , entre 0 e  $l - 1$ , onde  $l$  é o tamanho da *string*, e todos os alelos após este ponto  $k$  são trocados entre os genitores.
- *Crossover* em dois pontos (Figura 2.3(b)): Dois pontos  $k$  e  $j$ ,  $k \neq j$ , são escolhidos aleatoriamente entre 0 e  $l - 1$  e os alelos entre  $k$  e  $j$  são trocados entre os genitores.
- *Crossover* uniforme (Figura 2.3(c)): Todos os alelos são trocados aleatoriamente com uma certa probabilidade,  $p_i$ , chamada de probabilidade de troca. Normalmente o valor adotado para esta probabilidade é entre 0,5 e 0,9.
- *Crossover* uniforme baseado em ordem (Figura 2.4): Os métodos acima não são apropriados para problemas de busca que utilizem permutação, como o problema do caixeiro viajante, pois eles muitas vezes geram soluções inválidas para o problema, com genes de valores repetidos. Neste método, dois genitores,  $G_1$  e  $G_2$ , são selecionados e uma *string* binária é gerada aleatoriamente. Alguns genes do primeiro filho,  $F_1$ , são obtidos dos genes de  $G_1$  onde as posições da *string* binária é igual a 1. Os genes de  $G_1$  que estão na posição 0 da *string* binária são selecionados e ordenados na mesma ordem em que aparecem em  $G_2$ . Estes valores são inseridos nos genes restantes de  $F_1$ . O segundo filho é gerado de maneira similar, utilizando  $G_2$ .

### 2.3.5 Mutação

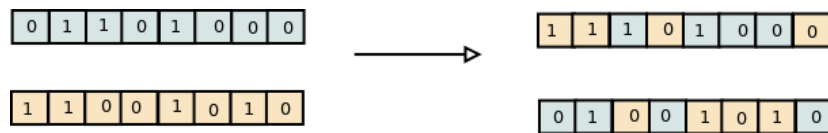
Ao contrário do *crossover* que ocorre entre dois genitores, a mutação ocorre em um só indivíduo. Através do *crossover* é possível obter soluções cada vez melhores, porém se os dois genitores possuírem o mesmo alelo em um determinado gene, este alelo não será



(a) *Crossover* em um ponto



(b) *Crossover* em dois pontos



(c) *Crossover* uniforme

Figura 2.3: Tipos de *crossover*.

modificado utilizando somente o *crossover* e este alelo nunca terá seu valor modificado. A mutação é utilizada para resolver este problema e adicionar diversidade a uma população e ela é implementada, geralmente, com uma pequena probabilidade. No caso de *strings* binárias, mutar um gene significa apenas trocar o valor do bit de 0 para 1 ou de 1 para 0 (Figura 2.5).

Muitos pesquisadores acreditam que a mutação é a parte mais importante dos algoritmos genéticos [19], mas para cada problema é importante encontrar o equilíbrio entre o *crossover* e a mutação.

### 2.3.6 Substituição

Após gerar uma nova população através do *crossover* e da mutação, esta nova população pode substituir a anterior. As seguintes técnicas são utilizadas para substituir a população dos genitores:

- Deletar todos: Esta técnica deleta todos os membros da população atual e os substituem pelos indivíduos que foram gerados nessa iteração do algoritmo. Essa é a técnica mais utilizada, devido a sua simplicidade e facilidade de implementação.



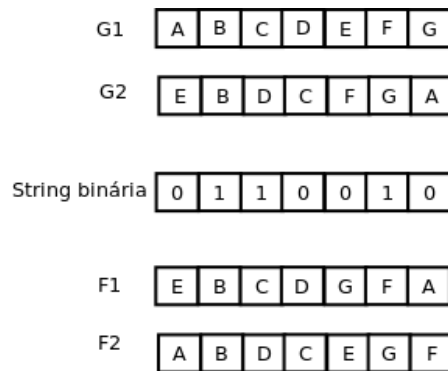


Figura 2.4: *Crossover* uniforme baseado em ordem.

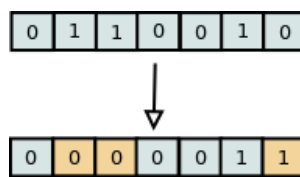


Figura 2.5: Um exemplo de mutação.

- Estado estável: Deleta  $n$  indivíduos e os substitui por outros  $n$  indivíduos gerados. É necessário definir um critério para se deletar os indivíduos, isto pode ser feito de forma aleatória ou se escolhendo os piores indivíduos.
- Estado estável sem duplicatas: É similar à técnica acima, porém neste caso a implementação deve garantir que não sejam inseridas strings duplicadas à população. Isto gera um *overhead* ao algoritmo, porém expande o espaço de busca.

### 2.3.7 Término da Execução

Os algoritmos genéticos são métodos de busca que poderiam ser executados eternamente sem encerrar, ou seja, ele não determina um ponto em que o algoritmo chega ao fim, por isso devem ser definidos critérios de paradas para o algoritmo. Alguns dos critérios comumente definidos levam em conta um valor de *fitness* pré-definido, ou um número de gerações, ou o tempo de *clock* do computador ou até mesmo contabilizando a diversidade da população e parando quando ela chegar a um limite e/ou parar de variar.

Finalmente, o algoritmo em sua forma mais geral para AG é apresentado a seguir.

- 1: escolha uma população inicial de cromossomos
- 2: **enquanto** condição de término não for satisfeita **faça**
- 3:     **repita**
- 4:         **se** condição do crossover for satisfeita **então**
- 5:             selecione os cromossomos genitores;
- 6:             escolha os parâmetros de crossover;
- 7:             execute o crossover;
- 8:         **fim se**
- 9:     **se** condição do mutação for satisfeita **então**

```
10:         escolha os pontos de mutação;
11:         execute a mutação;
12:     fim se
13:     avalie o valor de fitness dos descendentes gerados;
14:     até número de descendentes ser grande o suficiente;
15:     selecione uma nova população;
16: fim enquanto
```

## 2.4 Paralelismo em Algoritmos Genéticos

Os algoritmos evolucionários são naturalmente paralelos, o grau de paralelismo é diretamente proporcional ao tamanho da população usada na busca genética. Quando o tempo de avaliação da função,  $T_f$ , é muito maior que o tempo de comunicação  $T_c$  é possível utilizar um modelo simples de mestre-escravo, onde as avaliações do valor de *fitness* são distribuídas entre os diversos processadores e o restante das operações dos AG são executadas por um único processador[56]. Alguns modelos de paralelismo para algoritmos genéticos são apresentados a seguir, todos esses modelos buscam maximizar o paralelismo e diminuir a comunicação desnecessária entre processos.

### 2.4.1 Populações Globais com Paralelismo

Neste modelo a seleção da população para preencher a área de reprodução é realizada através da seleção por torneio. Supondo que os processadores são enumerados de 1 a  $N/2$  e o tamanho da população,  $N$ , é par, então 2 *strings* serão alocadas em cada processador[56]. Cada processador realiza dois torneios independentes, inserindo *strings* aleatórias da população nos torneios e cada processador guarda os vencedores dos dois torneios. As novas *strings* que estão armazenadas nos processadores representam a população selecionada. Desta forma o *crossover*, a mutação e a avaliação podem ocorrer de forma paralela.

### 2.4.2 Modelo de ilhas

Este é um modelo de paralelismo com maior granularidade (“*coarse grain*”). Supondo que sejam usados 16 processadores e uma população de 1.600 *strings*, uma maneira de se implementar o paralelismo é dividir a população total em subpopulações de 100 *strings* cada. Cada uma destas subpopulações executarão o algoritmo genético de forma independente. Ocasionalmente estas subpopulações podem trocar algumas *strings* para dividir o material genético e aumentar a diversidade, esta troca é chamada de migração. Se essa migração não ocorrer, cada busca será diferente, pois as populações iniciais vão direcionar as gerações para rumos diferentes.

O modelo de ilhas permite explorar as diferenças das várias subpopulações, fornecendo uma fonte de diversidade genética. Cada subpopulação representa uma ilha e deve existir uma maneira para o material genético ir de uma ilha para a outra. O número de *strings* que são passadas de uma ilha para a outra deve ser pequeno para evitar que as diferenças entre as ilhas diminuam demais. A Figura 2.6 é uma ilustração do modelo das ilhas, as

cores representam a similaridade do material genético, as ilhas mais distantes possuem menor similaridade.

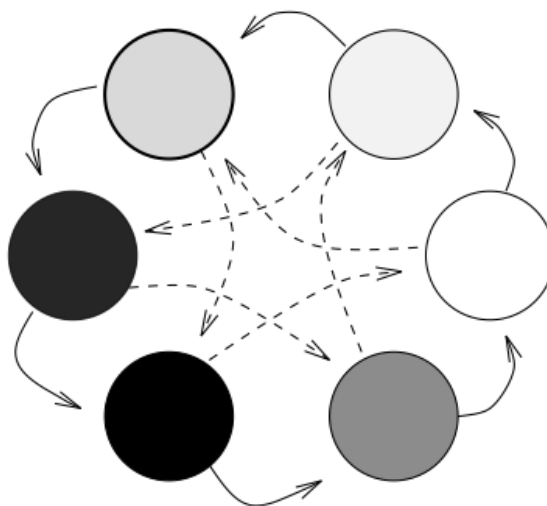


Figura 2.6: Modelo de ilhas[56].

## 2.5 Aplicações dos Algoritmos Genéticos

Os algoritmos genéticos são muitas vezes vistos como otimizadores de funções, mas eles têm sido aplicados em diversos tipos de problema. Neste seção são apresentadas duas aplicações dos AG, a hibridização[37] e uma aplicação na Teoria dos Jogos[47].

### 2.5.1 Hibridização

É um modo muito eficaz de melhorar o desempenho e eficácia dos algoritmos genéticos. A forma mais comum de hibridização é agrupar os algoritmos genéticos com técnicas de busca local e incorporar conhecimentos específicos do domínio ao processo de busca.

De acordo com L. “Dave” Davis[37], os algoritmos genéticos tradicionais, apesar de serem robustos, eles, geralmente, não representam a melhor técnica de otimização para nenhum domínio. Utilizar AG híbridos com alguma técnica eficiente para um problema específico, faz com que o melhor dos dois mundos seja aproveitado, apesar de ocorrer um aumento no *overhead*. Existem três princípios para a hibridização:

1. Deve se utilizar uma codificação que já seja conhecida e usada nos métodos usuais de solução do problema;
2. Hibridizar onde for possível, incorporando elementos de outros algoritmos que já foram experimentados na solução do problema;

3. Adaptar os operadores genéticos para que eles se tornem compatíveis com o problema;

## 2.5.2 Teoria dos Jogos com AG

Em um jogo matemático típico, algumas regras são estabelecidas e a recompensa para os jogadores depois de cada movimento é dada por uma função de *payoff*. Muitas vezes é necessário descobrir qual é a melhor jogada possível, ou seja, aquela em que o jogador consegue maximizar seu *payoff*. Em muitos jogos o efeito cumulativo da função de *payoff* não pode ser expresso analiticamente e a única possibilidade é realmente jogar o jogo e comparar as diferentes estratégias.

Os AG podem ser utilizados para simular estes jogos, uma população de jogadores é gerada e ocorre um torneio. Cada estratégia deve ser codificada como um cromossomo e passar por mutações e *crossover*. Em cada rodada do torneio, o *payoff* de cada estratégia é calculado e então os melhores jogadores fornecerão uma maior quantidade de genes para as próximas gerações[47].

O problema do dilema do prisioneiro pode ser otimizado com o uso de algoritmos genéticos[48]. Neste problema, que envolve dois jogadores, cada jogador deve decidir se ele quer cooperar com o outro jogador ou traí-lo. Existem 4 possibilidades:

1. Os dois jogadores cooperam;
2. O primeiro coopera e o segundo trai;
3. O primeiro trai e o segundo coopera;
4. Ambos traem um ao outro;

		Primeiro Jogador	
		C	T
Segundo Jogador	C	3, 3	0, 5
	T	5, 0	1, 1

Figura 2.7: Matriz de *payoff* do dilema do prisioneiro. A letra C representa cooperação e a letra T representa traição.

A Figura 2.7 mostra os valores de *payoff* para cada estratégia, é observado que é vantajoso cometer traição. O *payoff* quando um jogador trai o outro, que quer cooperar, é 5 e o jogador que cooperou não ganha nada. Se ambos os jogadores traírem, ambos recebem 1. Se os dois cooperarem, ambos recebem 3. Se o jogo for jogado só uma vez, a melhor estratégia é trair. Mas o jogo fica mais complicado se for repetido diversas vezes, jogadores que desejam cooperar juntos conseguem um *payoff* melhor do que rodadas em que

ambos traem. Em sua pesquisa, Axelrod[48] realizou um torneio em que a população era formada por diferentes jogadores com diferentes estratégias, utilizando AG. Cada jogador podia armazenar o resultado dos últimos três jogos e os jogadores eram emparelhados de forma aleatória.

Como existem quatro possíveis resultados para o jogo e somente os três jogos anteriores eram armazenados, existiam 64 possíveis históricos recentes do jogo contra cada oponente. A estratégia de cada jogador pode ser codificada em um cromossomo binário de tamanho 64, onde cada componente representa um dos possíveis históricos e o valor 1 representa cooperação e o valor 0 representa traição, para cada histórico possível o jogador deveria escolher 0 ou 1. Descobriu-se que uma estratégia simples era a mais eficaz, a estratégia se chama “*tit for tat*”, ela consiste em apenas repetir o último movimento do oponente.

# Capítulo 3

## Conceitos de ATFM

O Brasil tem a responsabilidade de administrar o espaço aéreo territorial (8.511.965  $km^2$ ) e o espaço aéreo sobrejacente à área oceânica, que se estende até o meridiano 10° W, perfazendo um total de 22 milhões de  $km^2$  [1].

Para garantir que isto ocorra de forma segura, deve se conhecer:

1. a demanda de tráfego aéreo atual e futura;
2. a topografia e a infra-estrutura instalada;

Este capítulo apresenta os conceitos mais relevantes de Gerenciamento de Fluxo de Tráfego Aéreo (*Air Traffic Flow Management*) (ATFM) para situar o leitor no contexto da aviação brasileira e mundial. A seção 3.1 apresenta os conceitos básicos sobre gerenciamento de tráfego aéreo, nas Seções 3.2 e 3.3 as funções do Gerenciamento de Tráfego Aéreo e Gerenciamento do Fluxo de Tráfego Aéreo são apresentadas, respectivamente. As etapas do voo são apresentadas na Seção 3.4 e as restrições operacionais na Seção 3.5. Os atuais problemas no tráfego aéreo brasileiro são apresentados na Seção 3.6 e uma iniciativa para melhorar estes problemas é exibida na Seção 3.7.

### 3.1 Conceitos Gerais

Com o crescimento da malha aérea global, os voos que antes eram operados segundo as Regras de Voo Visual (*Visual Flight Rules*) (VFR) em um quadro que não apresentava problemas como congestionamentos, ou probabilidades de colisão pelo baixo volume de aeronaves e de voos [10], visando diminuir o número de incidentes e evitar falhas humanas, regras para execução de voos com auxílio de instrumentos foram estabelecidas por órgãos de controle, as Regras de Voo Instrumentado (*Instrumented Flight Rules*) (IFR).

#### 3.1.1 Serviços de Tráfego Aéreo - ATS

Responsável pelo relacionamento entre o operador de um órgão de tráfego aéreo e o piloto da aeronave, o Serviço de Tráfego Aéreo (*Air Traffic Service*) (ATS) deve prover recursos de comunicação e tecnologia que atendam as necessidades, como manutenção da distância mínima entre aeronave, disponibilidade das informações de voo, alerta em casos que necessitem resgate e outros.

As diversas modalidades de prestação de serviço prestadas pelos ATS [10] podem ser classificadas em:

1. Serviços de Informação de voo;
2. Serviços de Alerta;
3. Serviços de Auxílio Instrumental à Navegação;
4. Serviços de Controle de Tráfego Aéreo:
  - Controle de Área;
  - Controle de Aproximação;
  - Controle de Aerodromo.

O órgão de ATS tem a competência de tomar ações para garantir a qualidade do serviço, através da formalização de protocolos de operações e elaboração de procedimentos para análise e correção de falhas.

## 3.2 Gerenciamento de Tráfego Aéreo

O Gerenciamento de Tráfego Aéreo (*Air Traffic Management*) (ATM) tem como objetivo manter as aeronaves cumprindo os horários previstos de partida e chegada a partir de planos de voo bem definidos, e de maneira a atender todas as exigências operacionais de segurança [50] através da manutenção de um fluxo ótimo do tráfego aéreo, condicionado à capacidade do Controle de Tráfego Aéreo (*Air Traffic Control*) (ATC) e as limitações impostas pelas autoridades provedoras de ATS [10]. A capacidade de um ATC pode ser descrita como quantidade de movimentos aéreos simultâneos que um setor de controle de tráfego aéreo comporta [16].

Os órgãos de ATC devem operar com precisão, pois neles está confiada a tarefa de prevenção de incidentes no tráfego aéreo [18], este controle pode ser dividido em controle terrestre e controle aéreo.

A principal tarefa de um ATM é definir estruturas, procedimentos e regras de utilização do espaço aéreo, tendo em vista a demanda de tráfego aéreo atual e futura na topologia existente [33]. O ATM é responsável por definir:

- as necessidades operacionais para implantação de órgão de controle de tráfego aéreo;
- estruturas para uso eficaz do espaço aéreo como, aerovias, procedimentos de pouso e decolagem, delimitar áreas controladas, não controladas e condicionadas etc;
- ações possíveis para cada seguimento do espaço aéreo;
- os espaços onde os controladores poderão prover a separação das aeronaves.

### 3.3 Gerenciamento do Fluxo de Tráfego Aéreo

O Gerenciamento de Fluxo de Tráfego Aéreo (*Air Traffic Flow Management*) (ATFM) é um ATS, provido com objetivo de controlar a fluência da malha aérea nas regiões onde o serviço é prestado. Espera-se que os horários de decolagem e aterrissagem sejam cumpridos de acordo com o plano de voo previsto.

Esta situação introduz complicações ao transporte aéreo pois deve ser garantido que o voo decole e pouse com segurança e com o menor atraso possível. Para que este objetivo seja alcançado, são eventualmente adotadas medidas restritivas como a diminuição da velocidade em voo ou esperas em ar (*Airborne Holding Points*) ou ainda desvios de rotas.

A distribuição das medidas restritivas de forma justa é uma das dificuldades do ATFM pois tais restrições de movimentação entre os usuários da malha aérea são acompanhadas de um custo a este usuário. Por custos pode se entender, o impacto causado por atrasos e por deslocamentos não previstos nos planos de voo. É fato que os atrasos em voos são acompanhados de um alto custo financeiro em termos de consumo de combustíveis [2], assim sendo qualquer alteração do plano de voo, é associado a um custo financeiro.

O serviço de Gerenciamento do Fluxo de Tráfego Aéreo possui o papel da busca por um fluxo ótimo de tráfego aéreo em uma situação de demanda superior à capacidade, através da aplicação de medidas restritivas [50].

Esta atividade é executada em fases que antecedem o voo, são elas:

- Planejamento estratégico: inicia-se 6 meses antes do voo e encerra-se na antevéspera, é a fase de coordenação das atividades relacionadas ao balanceamento entre demanda e a capacidade, tem foco no planejamento de ações a serem tomadas em casos de congestionamentos.
- Planejamento pré-tático: tem início na véspera e se encerra até duas horas antes do voo, são realizadas projeções de demanda junto aos órgãos de controle sendo consideradas alterações na infraestrutura aeronáutica e aeroportuárias e verificadas as condições meteorológicas.
- Planejamento tático: tem início duas horas antes do voo e se encerra na aterrissagem. Compreendem atividades de monitoramento das atividades do voo em acordo com o ATC [17] e do espaço aéreo para verificar a possível ocorrência de eventos não previstos.

### 3.4 Fases do Voo

Um voo que siga as regras de voo instrumentado IFR, é dividido em fases que perpassam desde o planejamento do voo até a completude do mesmo. As etapas são divididas segundo [2] em:

- Pré-voo: É a fase foco deste trabalho, é compreendida pelas fases de planejamento pré-tático e planejamento tático descritos na seção 3.3.
- Taxiamento (*taxi out*): após receber autorização, a aeronave realiza o *push back*, que coloca a aeronave na pista de rolagem, o piloto então aciona os motores e conduz a aeronave até a cabeceira da pista e aguarda por confirmação para decolar.



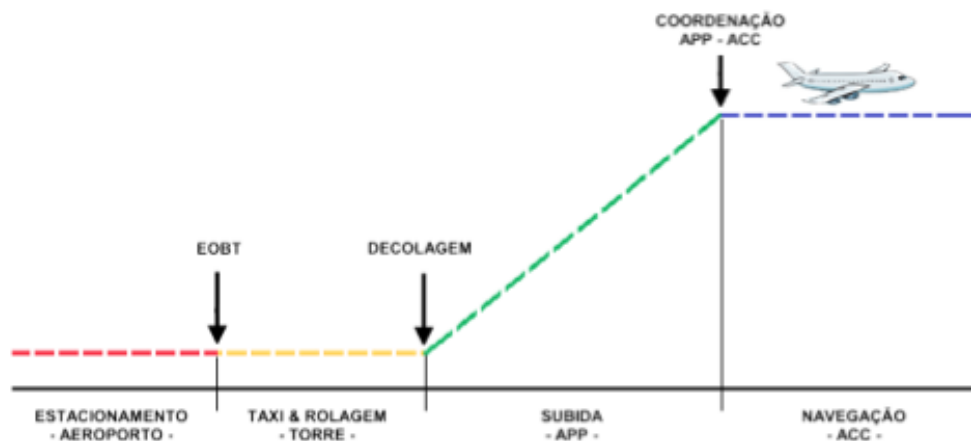


Figura 3.1: Processos de decolagem [25].

- Decolagem: após a autorização da torre, a aeronave pode decolar, ao atingir a altitude de 35 pés, a aeronave deixa o Área de Controle de Torre (*Tower*) (TWR) e passa para o Controle de Aproximação (*Approximation*) (APP), que se responsabilizará por guiar a aeronave na subida dentro da Área Terminal de Controle (TMA).
- Ascendência: processo contínuo de subida até a altitude de cruzeiro prevista no plano de voo. Ao atingir a altitude especificada em carta de rota (15000 pés em média), a aeronave sai da TMA e vai para a aerovia determinada.
- Cruzeiro: a aeronave ajusta sua altitude e velocidade e passa a ser responsabilidade do Centro de Controle de Área (*Area Control Center*) (ACC).
- Descida inicial: a aeronave deve reduzir a altitude até atingir a altura da TMA. Em caso do terminal estar congestionado, a aeronave passa por medida restritiva para que sua entrada na TMA seja atrasada, o que acarreta em um aumento do custo do voo [38].
- Aproximação final: Ao chegar ao terminal de destino, a aeronave volta ao controle do APP, é corrigida a velocidade de descida e a aterrissagem é autorizada pela torre de controle.
- Taxiamento (*taxi in*): após tocar o solo, a aeronave reduz a velocidade para a velocidade de taxiamento, de acordo com o modelo de aeronave, e é conduzida ao portão de desembarque. A equipe de controle de solo é responsável por conduzir a aeronave pelo pátio sem interferir com os demais pousos e decolagens.

## 3.5 Restrições operacionais

### 3.5.1 Controle de saturação

O gerenciamento de tráfego em um aeroporto é feito pelos controladores de torre que dividem essa tarefa em três: alocação(sequenciamento), taxiamento e controle de pista

afim de diminuir a complexidade [45].

A torre de controle não pode liberar mais voos para decolar que o APP comporta e o aeródromo não pode receber mais aeronaves do que sua capacidade instalada, assim a chegada e a partida de aeronaves sofre dupla restrição tanto do APP quanto da capacidade de armazenamento do aeroporto.

O procedimento de aterrissagem tem prioridade sobre a partida, isso faz com que as decolagens sejam encaixadas entre os pousos, pois o consumo de espera no ar é mais elevado que em solo. O aeroporto atinge um grau de saturação quando o elevado número de aterrissagens supera o número de decolagens, quando isso ocorre é necessário interromper as aterrissagens para que as decolagens possam ocorrer, em caso de aeronaves em velocidade de cruzeiro, sua velocidade, nível de voo ou rota são alterados.

O congestionamento de um setor, rota de uma aeronave, ocorre quando a ocupação do setor chega a 80% de sua capacidade e a saturação ocorre quando o setor atinge sua capacidade máxima, que nunca deve ser ultrapassada pelo risco de colisões.

### 3.5.2 Procedimentos pré-decolagem

O gerenciamento de pista visa o agendamento eficiente das decolagens, através da elaboração de um plano de decolagem para cada aeronave. Este plano é composto por:

- planejamento de *gate*;
- plano de taxiamento;
- plano de *runway*.

O primeiro é responsável pela designação de posição de início em solo, duração do embarque de passageiros e carga; o segundo é responsável pelo deslocamento da aeronave até a pista de rolagem e o último pela condução da aeronave até a cabeceira da pista [45].

O táxi é definido pelo Ministério da Defesa como o movimento autopropulsado de uma aeronave sobre a superfície de um aeródromo, excetuando-se os instantes de decolagem ou aterrissagem [39]. O táxi ocorre sempre nas decolagens e nas aterrissagens, o primeiro é chamado de *taxi out* e o segundo de *taxi in*. Os escopo deste trabalho engloba o *taxi out*. No processo de decolagem a aeronave solicita autorização antes de ligar os motores, é então definido a Meta de Horário para Decolagem (Target Time for Departure) (TDD), que é o horário previsto para a decolagem e a Meta de Tempo para Início de Movimentação (*Target Off-Block Time*) (TOBT) que é o momento em que a mesma deixa a posição de embarque e inicia a movimentação de solo.

O papel do controlador de solo é ser responsável pelo tráfego da aeronaves durante o período de *taxi out* e uma de suas tarefas é evitar que:

- Aeronaves utilizem a mesma rota em direções e sentidos opostas (Cruzamento);
- Que duas aeronaves utilizem a mesma rota no mesmo sentido e que a segunda esteja com uma velocidade maior que a primeira;
- Que duas aeronaves trafeguem uma em direção à outra (colisão frontal)

Para os cálculos as informações do Horário Estimado de Parada Total (*Estimated In-block Time*) (EIBT), do Horário Estimado de Decolagem (*Estimated Takeoff Time*) (ETOT) e do Horário Calculado de Decolagem (*Calculated Takeoff Time*) (CTOT), e que se conheça o tempo de taxiamento de cada aeronave é importante. Esse tempo pode variar de acordo com cada aeroporto [24].

Os fatores que podem afetar o tempo de taxiamento são, entre outros, a infra-estrutura aeroportuária, as pistas em utilização, condição meteorológicas, modelo e peso da aeronave, tráfego e procedimentos operacionais.

O cálculo de período de taxiamento em aeroportos grandes, depende do cálculo do Horário Estimado de Taxi-In (*Estimated taXi-In Time*) (EXIT) e do Horário Estimado de Taxi-Out (*Estimated taXi-Out Time*) (EXOT). Os principais fatores levados em consideração para este cálculo são:

- os tempos padrões de taxiamento;
- os dados de movimentação das aeronaves no TMA;
- tipo e categoria das aeronaves;
- rotas de taxiamento;
- histórico de taxiamento;
- tempos específicos de taxiamentos.

### 3.5.3 Separação entre aeronaves

A separação entre aeronaves em solo é importante para que as manobras realizadas por uma aeronave não afete ela mesma ou outras aeronaves e seus tripulantes. Dois momentos se destacam quanto a separação entre aeronaves em solo.

- durante o sequenciamento de aproximação;
- nas aterrissagens e decolagens.

As distâncias mínimas adotadas visam agilizar o tráfego e manter a segurança dos voos, por isso são adotadas medidas para evitarem a esteira de turbulência, que são vórtices geradas pelas aeronaves pelo contato das asas com o vento, para que a separação entre as aeronaves possa ser calculada são considerados os seguintes parâmetros:

- peso da aeronave;
- velocidade;
- configuração.

## 3.6 O Tráfego Aéreo no Brasil

O atual sistema ATM brasileiro apresenta diversos problemas, alguns deles são [41]:

- Incompatibilidade de sistemas e procedimentos, devido ao uso de sistemas/ferramentas distintas;

- Sistemas limitados para suporte à decisão;
- Dependência de comunicações ar-terra em voz, que estão cada vez mais congestionadas;
- Rotas rígidas no espaço aéreo, que não permitem utilizar ao máximo os recursos ATM;
- Limitado planejamento colaborativo entre operadores de aeroporto, operadores de aeronaves e provedor de gerenciamento de tráfego aéreo;
- Uso não otimizado da capacidade do espaço aéreo e da parte aeronáutica dos aeroportos;
- Infraestrutura limitada para troca de informações em tempo real, o que impossibilita obter respostas quando há eventos e modificações nos requisitos operacionais dos usuários;
- Não se aproveita ao máximo os avançados sistemas de bordo das aeronaves;
- Recursos humanos insuficientes em qualidade e quantidade;

### 3.7 CSM/ATM

Em 1991 nasceu um conceito desenvolvido para atender as necessidades previstas para a aviação do século XXI, esse conceito foi chamado de CNS/ATM e foi oficializado pela OACI(Organização de Aviação Civil Internacional) na 10ª Conferência de Navegação Aérea [52].

O CNS/ATM procura aplicar tecnologia de satélites, comunicação digital e gestão estratégica das operações, com integração de tecnologias, processos e recursos humanos que sejam capazes de suportar a evolução do transporte aéreo. A sigla CNS/ATM é composta por 4 expressões: Comunicação Aeronáutica(letra C), Navegação Aérea(letra N), Vigilância Aérea(letra S, de *Surveillance*) e Gerenciamento de Tráfego Aéreo(ATM).

No Brasil, foi elaborada pelo Departamento de Controle do Espaço Aéreo (DECEA) uma concepção específica para o processo brasileiro em 2008: a Concepção Operacional ATM Nacional (DCA-351-2) [41]. O plano de implementação é composto por três fases:

- Fase 1 - curto prazo - até 2010: aplicação de procedimentos, processos e capacidades disponíveis;
- Fase 2 - médio prazo - de 2011 até 2015: aplicação de procedimentos, processos e capacidades emergentes;
- Fase 3 - longo prazo - de 2016 até 2020: imigração evolutiva ao sistema ATM Global;

Esse modelo trará diversos benefícios para a aviação brasileira, alguns deles são [42]:

1. Uso racional do espaço aéreo;
2. Redução da emissão de gases nocivos na atmosfera;
3. Redução de ruído em locais próximos a aeroportos;

4. Redução da carga de trabalho dos controladores;
5. Redução da carga de trabalho dos pilotos;
6. Redução de custos para provedores dos serviços de navegação aérea;
7. Redução de custos para os operadores de aeronaves;
8. Melhor atendimento na prestação do transporte aéreo para os usuários;
9. Maiores investimentos em tecnologia de comunicações e segurança;

Ainda há um longo caminho até a implementação completa do modelo, mas algumas iniciativas já estão em curso:

1. Criação do Centro de Gerenciamento da Navegação Aérea (CGNA).
2. Aplicação da tecnologia de Vigilância Aérea ADS-C no Centro de Controle de Área Atlântico (ACC-AO), incorporando benefícios de segurança e eficácia ao fluxo de tráfego aéreo sobre o oceano.
3. Implantação da Navegação Baseada em Performance (PBN) nos terminais de Recife e Brasília.
4. Instalação de dispositivos que propiciam a aproximação de precisão por satélites, o GBAS (Sistema de Aumentação Baseado em Solo), no Aeroporto do Galeão, para a realização dos primeiros testes operacionais em 2011.
5. Início das pesquisas para a implementação do Sistema ADS-B (Vigilância Dependente Automática por Radiodifusão), nas operações *off shore* da Baía de Campos.
6. Implantação - em andamento - de ferramenta de sequenciamento de aeronaves em área terminal (rota de chegada e aproximação).

# Capítulo 4

## Collaborative Decision Making - CDM

O CDM é uma iniciativa de governos e indústrias para melhorar o gerenciamento de tráfego aéreo[23]. É formado por representantes do governo, empresas aéreas, indústria privada e acadêmicos que trabalham em conjunto para criar novas tecnologias e procedimentos que possam ser utilizados como soluções para os problemas do tráfego aéreo.

Este capítulo apresenta primeiramente, na seção 4.1 o funcionamento e os objetivos do CDM. A seção 4.2 apresenta como é feita a alocação inicial de *slots*, seu histórico e sua aplicação no contexto brasileiro. Nas seções 4.3 e 4.4 são apresentados os algoritmos RBS e *Compression*, respectivamente, que são utilizados para alocar os *slots* quando ocorrem atrasos ou cancelamentos.

### 4.1 Funcionamento

O CDM[22] é um paradigma onde as decisões de gerenciamento de tráfego aéreo se baseiam nos princípios de compartilhamento de informação e na tomada de decisões distribuídas[40].

Todos os aeroportos participantes do CDM possuem informações atualizadas e, com isso, uma visão global de toda a situação do tráfego aéreo, permitindo melhorar seus processos de planejamento. Para que exista uma transparência no tráfego aéreo entre os participantes do CDM, alguns requisitos são necessários[25]:

1. Os dados que forem relevantes, deverão ser divididos quando necessário(Figura 4.1).
2. Os dados compartilhados devem ser suficientes para melhorar a capacidade de planejamento e facilitar a gerência de trafego aéreo para todos os parceiros.
3. As decisões devem ser realizadas pelo parceiro em melhor condições de realizá-la.
4. As decisões feitas devem ser reportadas aos outros parceiros.

Os objetivos do CDM são:

- Gerar melhores informações, geralmente mesclando informações do Sistema do Espaço Aéreo(NAS) com informações geradas por usuários do espaço aéreo;
- Criar uma consciência comum da situação, distribuindo a mesma informação, tanto para quem gerencia o tráfego aéreo quanto para os usuários do espaço aéreo;

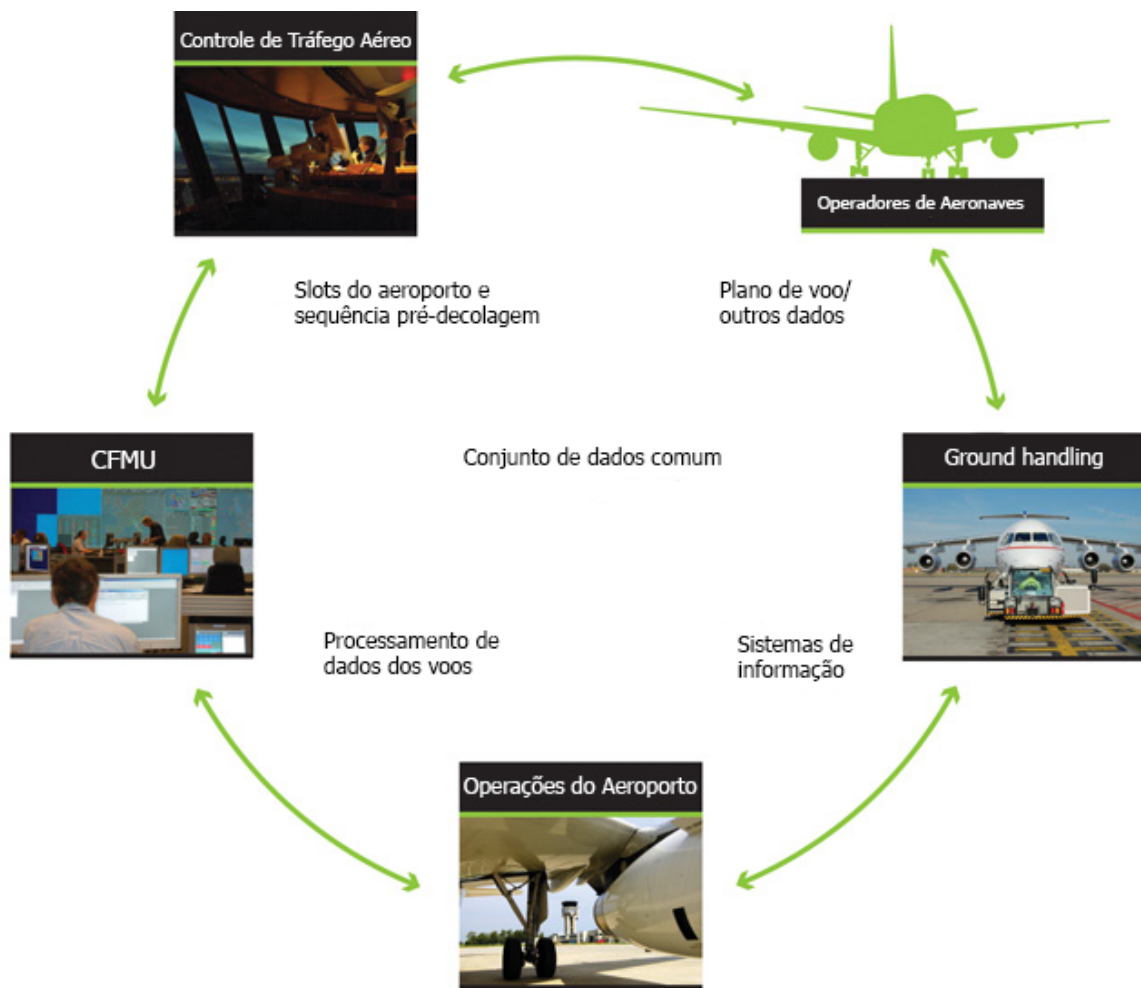


Figura 4.1: Compartilhamento de informações nos aeroportos da Europa que utilizam o CDM[25].

- Criar ferramentas e procedimentos que permitam que os usuários do espaço aéreo respondam diretamente aos desequilíbrios de capacidade/demanda e colaborem com o gerenciamento de tráfego aéreo;

Os custos de utilizar o CDM não são altos, ele não introduz nenhuma novidade radical, o que deve ser realizado é uma melhoria nos processos e sistemas já existentes. Existem diversas aplicações para o CDM e elas são divididas em quatro níveis:

- Primeiro nível: O objetivo principal é criar a consciência situacional comum (*common situational awareness*) e melhorar a previsibilidade do tráfego.
- Segundo nível: Melhorar a pontualidade, inserindo uma flexibilidade operacional, capaz de se adaptar a mudança de tráfego e a preferência do operador.
- Terceiro nível: Aumentar ainda mais a flexibilidade e otimizar os recursos do aeroporto e torná-los capaz de se recuperar em situações extremas, como a ocorrência de condições climáticas adversas.

- Nível avançado: Novas aplicações que surgirem, construídas em cima das já existentes.

O CDM é utilizado principalmente na Europa e nos EUA, porém surgiram em contextos diferentes, nos EUA ele surgiu para se lidar com a redução das capacidades dos aeroportos, devido ao clima ruim durante os voos e nos aeroportos. Já na Europa, o CDM focou em aumentar a previsibilidade das operações dos aeroportos para evitar que os mesmos se tornassem o gargalo do sistema de gerenciamento de tráfego aéreo. O CDM foi inicialmente utilizado em 1998 no aeroporto de São Francisco com sucesso e então aplicado a todos os aeroportos dos EUA[34].

Sem o CDM, decisões operacionais muitas vezes podem estar incorretas, ou nem chegarem a ser tomadas. Muitos parceiros tomam decisões incorretas devido a falta de informação ou falta de um padrão, que torna essa informação ambígua ou com significado incorreto para alguma das partes. Uma decisão de gerenciamento de tráfego aéreo feita em termos locais, irá ser influenciada por eventos que aconteceram a centenas de quilômetros de distância.

Estudos realizados pela Organização Europeia para a Segurança da Navegação Aérea (*European Organization for the Safety of Air Navigation*) (Eurocontrol)[25] já demonstraram que houve um aumento de 15% no número de voos que decolaram na hora marcada com a implementação do CDM. O CDM já provou ser aplicável em diversos problemas do gerenciamento de tráfego aéreo, como congestionamento do espaço aéreo devido ao tráfego intenso e para fazer com que as aeronaves evitem rotas com o clima severo. Para esse paradigma é fundamental o surgimento de novos princípios de alocação, que são incorporados aos algoritmos de *Compression* e de RBS[27].

## 4.2 Alocação de Slots

O *slot* é definido como o horário ou período de tempo estabelecido para uma aeronave realizar uma operação de chegada ou uma operação de partida em um aeroporto coordenado[46]. A utilização dos *slots* é que vai determinar a sequência de partidas nos aeroportos, sendo assim, é fundamental para este trabalho conhecer como é feita a distribuição e alocação de *slots*.

### 4.2.1 Histórico

Desde o início dos voos comerciais, o direito de decolar e aterrissar foi alocado de acordo com uma política de *first-come first-served* (primeiro que chega é o primeiro a ser servido). Com a chegada dos anos 60, as filas de aviões esperando para utilizar as pistas de decolagem ou aterrissagem se tornaram comum nos maiores aeroportos internacionais.

Para diminuir os atrasos, em 1968 a FAA determinou que o número de *slots* fosse limitado nos quatro maiores aeroportos dos Estados Unidos. Os *slots* eram divididos através de acordos entre as empresas aéreas, sem necessidade de uma alocação eficiente e justa dos *slots*. Com o crescimento no número de companhias aéreas a necessidade de se regular a alocação de *slots* ficou evidente. Em 1985, a FAA apresentou novas regras para a alocação de *slots*, baseadas na precedência histórica e permitindo a comercialização dos



*slots*. Na Europa, a alocação de *slots* é regulada desde 1993 e não se permite que os *slots* sejam trocados em transações monetárias.

Os *slots* são alocados gratuitamente para as companhias aéreas, elas devem apenas pagar uma taxa ao aeroporto pelo uso do mesmo. Neste mecanismo, os aeroportos são considerados independentes uns dos outros, não se leva em conta o fato de que, para cada voo, uma companhia aérea precisa de uma combinação de *slots* nos aeroportos de origem e destino.

#### 4.2.2 Alocação de *Slots* no Brasil

A alocação de *slots* no Brasil é regulamentada pela Agência Nacional de Aviação Civil (ANAC), a regulação de *slots* foi a primeira matéria a ser objeto de regulamentação específica, com a edição da Resolução nº 2 da ANAC. Essa resolução tratava apenas dos *slots* disponíveis, ou seja, aqueles que não competem a nenhuma empresa. As empresas incumbentes tem o direito de permanecer com seus *slots* por tempo indefinido, respeitando as condições de aproveitamento. Essa situação representa o princípio chamado de *grandfather rights*, que diz que se uma empresa aérea usou um *slot* no último verão/inverno, ela pode usá-lo no verão/inverno atual[28]. A empresa deve ter um mínimo de aproveitamento, para evitar que ela mantenha o *slot* apenas por questões de competição e não o utilize, isso pode ser descrito também pelo princípio *use it or lose it*(use-o ou perca-o).

A Resolução 2/2006 exige que as empresas aéreas tenha uma utilização trimestral de um percentual mínimo de 80% para que elas possam manter um determinado *slot*. Além disso, a empresa não deve interromper o uso do *slot* por mais de trinta dias contínuos. Os *slots* disponíveis surgem devido a empresas que não obedecem essas regulações de uso ou que até mesmo devolvem o *slot* por falta de interesse. Estes *slots* são redistribuídos, 80% passam para empresas que já atuam no aeroporto e 20% para empresas novas. A venda de *slots* é expressamente proibida e a Resolução 2/2006 não prevê mecanismos de mercado para a alocação inicial dos *slots*.

Esse modelo de alocação de *slots* acaba limitando o acesso de novas empresas. No Aeroporto de Congonhas as empresas TAM e GOL possuem, juntas, mais de 93% do total de *slots* dos aeroportos. Isso restringe a concorrência e pode fazer com que os passageiros acabem pagando passagens mais caras[11]. As regras de uso não costumam ser suficientes para que os *slots* sejam utilizados de forma eficiente, pois esses 80% de uso mínimo ainda permitem que as empresas não utilizem alguns *slots* e ainda sim os mantenham. A Figura 4.2 mostra a distribuição da participação no mercado das principais empresas aéreas em 2011.

Manter o *slot*, mesmo que com um uso mínimo, é benéfico para as empresas, que conseguem receitas adicionais limitando a competição. A resolução da ANAC exclui da regra de "*use it or lose it*" os voos cancelados por sazonalidade da operação ou feriado, além de motivos de força maior. O gráfico da Figura 4.3 mostra a distribuição de *slots* entre as empresas nos últimos anos, é possível perceber a dominância da TAM e Varig até 2006 e, mais recentemente TAM e GOL.

Existe ainda a alocação de *slots* que acontece próximo do tempo dos voos, que ocorre quando um *slot* fica vazio. Os dois algoritmos mais utilizados para esta alocação são o RBS e o *Compression*.

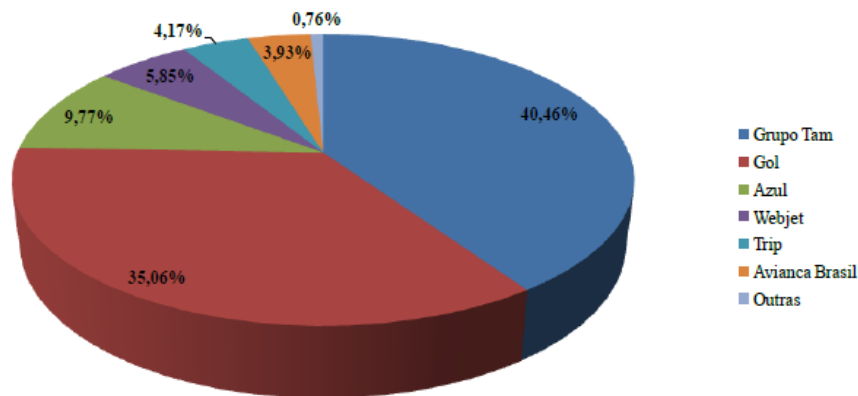


Figura 4.2: Participação no mercado das empresas aéreas em 2011[4].

### 4.3 RBS - *Ration-by-Schedule*

O RBS é um esquema de alocação de *slots* em que se segue a seguinte ideia "primeiro agendado, primeiro atendido", ou seja, os *slots* de chegada são alocados baseando-se no horário de chegada original agendado pelo voo, em vez do horário de partida no dia de operação. O planejamento resultante não é tão eficiente, pois não modifica a atribuição de *slots* quando ocorrem atrasos/cancelamentos[55]. Quando os voos são cancelados ou atrasados, cada companhia aérea retém seus direitos sobre os *slots* de chegada e podem designar, elas mesmas, novos voos para estes *slots* de acordo com seu planejamento[8].

O algoritmo RBS não diferencia o tamanho do *slots*, sendo assim, qualquer voo pode ser designado para qualquer *slot*. Os passos desse algoritmo são[12]:

#### Passo 1

Determinar o tempo dos *slots* de acordo com a taxa de aceitação máxima do aeroporto.

#### Passo 2

Sequenciar todos os voos de acordo com o seu tempo de agendamento de chegada.

#### Passo 3

Designar o próximo voo na sequência para o *slot* que estiver disponível o mais cedo possível, que estiver exatamente no tempo agendado para o voo, ou mais tarde.

#### Passo 4

Repetir o passo 3 até acabarem os voos.

O CDM utiliza esse algoritmo, junto ao *Compression*, para ajudar na alocação de *slots* quando ocorrem atrasos de voos e a realocação é necessária. O problema deste algoritmo é o fato do tamanho dos *slots* serem iguais, pois isso muitas vezes não reflete a realidade

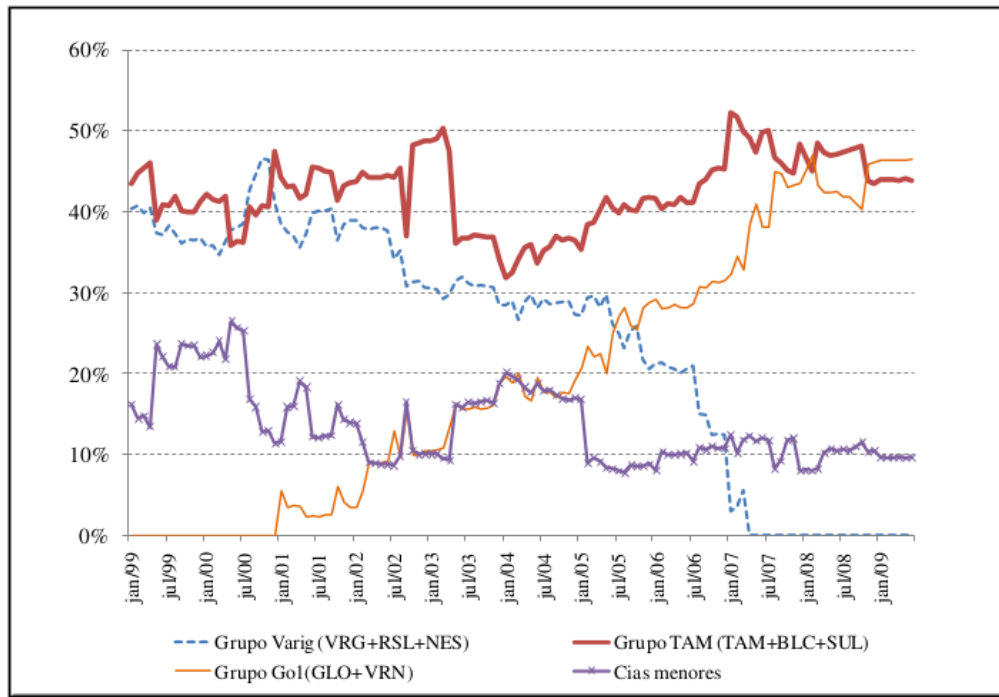


Figura 4.3: Dados dos *slots* de cada empresa aérea[46].

dos aeroportos. O tamanho dos *slots* irá depender da sequência de voos e de outras características operacionais.

Além disso, o *slot* pode ficar vazio se a empresa aérea não puder ou preferir não usá-lo.

## 4.4 Compression

O *Compression*[54] é um algoritmo que visa melhorar a utilização de *slots*, que por vezes devido a cancelamentos ou atrasos podem causar "buracos" na agenda, fazendo com que voos melhorem sua posição avançando suas posições na agenda de voos. A idéia por de trás do *Compression* é a de recompensar empresas aéreas que liberam seus *slots* e como consequência as empresas aéreas são encorajadas a informar cancelamentos. O escopo de aeronaves que podem ser movidas para cima na agenda de voos é limitado, já que um voo não pode decolar muito tempo antes do seu horário de decolagem agendado. O procedimento também leva em consideração o fato que voos não podem ser rebaixados da sua posição atual na agenda de decolagem  $I$ . A posição de *slots*  $\{e_j, \dots, I(f_j)\}$  que por consequência especificam a janela em que o voo  $f_i$  pode aterrissar.

O algoritmo de *Compression* pode ser resumido da seguinte forma:

### Passo 1:

Organizar os vôos de acordo com a agenda original. Determinar o número de *slots* abertos  $C_S$ .

## Passo 2:

Determinar o proprietário do *slot*  $c$ , que é, a linha aérea  $a$  que é proprietária do voo cancelado ou atrasado  $f_i$ , que foi alocado no *slot*  $c$ . Tentar alocar o *slot*  $c$  de acordo com as seguintes regras:

1. Determinar o primeiro voo  $f_i$  da aeronave  $a$  (na agenda atual) que pode ser alocado para o *slot*  $c$  para o qual  $c \in \{e_j, \dots, I(f_j)\}$ . Se não existir tal voo, o algoritmo irá para o passo 2.2 caso contrário para o passo 3.
2. Determinar o primeiro voo  $f_i$  de qualquer outra empresa aérea que pode ser alocado no *slot*  $c$ . Se não houver tal voo, o algoritmo irá para o passo 2.3, caso contrário para o passo 3
3. Não há voo que possa ser alocado no *slot*  $c$ . O algoritmo deve retornar ao Passo 1 e selecionar o próximo *slot* aberto.

## Passo 3:

Troque a alocação do *slot*  $f_i$  e  $f_j$ , por exemplo, aloque o voo  $f_j$  para o *slot*  $c$  e o voo  $f_i$  para o *slot*  $I(f_j)$ . Note que a linha aérea  $a$  agora é proprietária do *slot* aberto  $I(f_j)$ . O *slot*  $I(f_j)$  é feito o *slot* atual e o passo 2 é repetido.

A Figura 4.4 mostra um exemplo de utilização do *Compression*. O voo AAL290:1601 foi cancelado e com a alocação inicial, seu *slot* ficou vazio. Com a utilização do *Compression* não foi possível alocar esse *slot* para o outro voo da empresa empresa AAL, porém foi possível realocar um de seus voos para uma posição melhor da fila de *slots* e voos de outras empresas também foram realocados para preencher os "buracos".

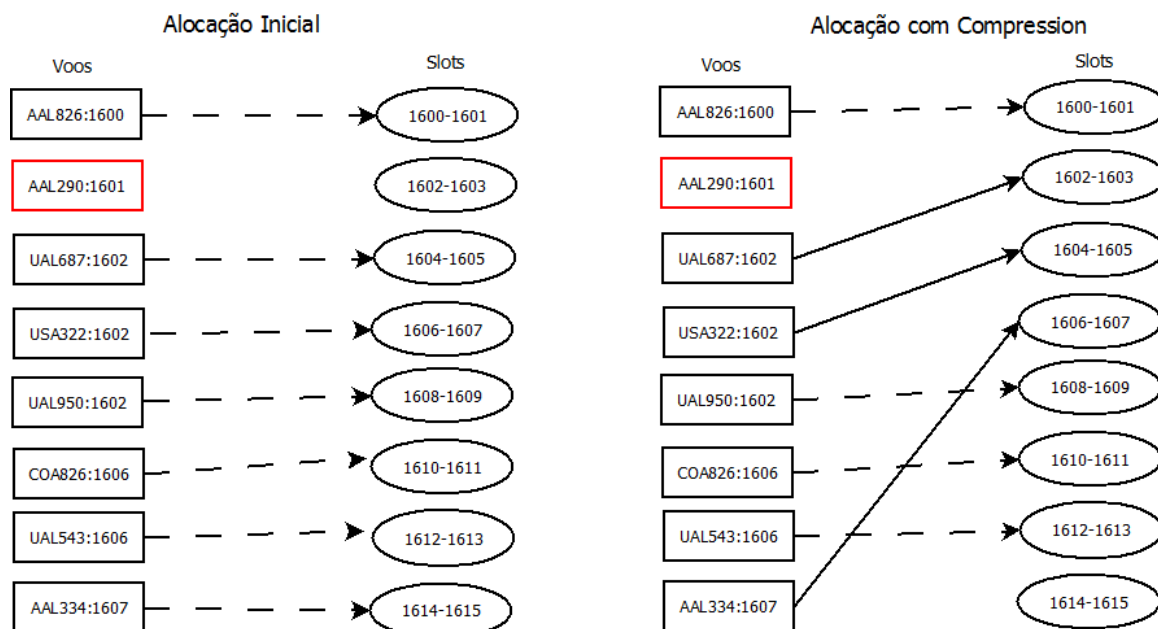


Figura 4.4: Exemplo de alocação utilizando *Compression*.

# Capítulo 5

## Estado da Arte

Para que se possa criar uma nova solução para o sequenciamento de partidas em aeroportos, é necessário primeiramente conhecer as soluções já existentes e verificar o que falta ser melhorado e os problemas ainda não tratados. O TransLab já realizou diversas pesquisas na área de melhorias no gerenciamento do transporte aéreo, um destes trabalhos foi o CoDMAN[53], que utiliza a Teoria dos Jogos para resolver o problema de Sequenciamento de Partidas, este trabalho é apresentado na seção 5.1. Na seção 5.2 é apresentado um trabalho sobre um sistema de gerenciamento de partidas que se baseia na ideia de criar uma fila virtual para economizar combustível e minimizar a emissão de poluentes diminuindo o tempo de táxi.

Na seção 5.3, uma solução para a alocação de *slots* em condições adversas é mostrada e, por último, uma demonstração das melhorias obtidas com o uso de Áreas de Absorção é apresentada na seção 5.4.

### 5.1 Alocação de *Slots* utilizando Teoria dos Jogos

A fim de tratar o problema do gerenciamento de partidas em aeroportos foi proposta uma pesquisa de uma abordagem experimental pelo TransLab, onde cada agente no modelo é uma aeronave que gera impacto nos cenários das movimentações aéreas, a este modelo foi dado o nome de **Gerenciamento Colaborativo de Partidas**(*Collaborative Departure Management*) (CoDMAN)[53].

Antes da decolagem são atribuídos às aeronaves atrasos imprevistos, garantindo assim a segurança e a eficiência operacional do tráfego aéreo. Os *slots* previamente alocados são alterados por esses atrasos o que força a manipulação da fila de partida. O CoDMAN propõem uma sequência eficiente de decolagens, fundamentada na negociação entre agentes em um cenário dinâmico desenvolvido com Teoria dos Jogos e Tomada de Decisão Colaborativa(CDM).

No mundo real uma aeronave quer o melhor *slot* de decolagem possível, isso cria um cenário predominantemente não-cooperativo. Em um espaço com  $n$  agentes que tentam maximizar sua eficiência, a teoria dos jogos pode ser aplicada.

### 5.1.1 Teoria dos Jogos no Sistema CoDMAN

O foco deste estudo[53] foi o processo de sequenciamento de aeronaves em aeroportos e o trabalho propôs a utilização de Teoria dos Jogos na fase de planejamento tático da realização de um voo, distribuindo de forma coordenada os *slots* existentes, conforme a demanda. Esse jogo deverá propiciar aos participantes informação perfeita, ou seja, cada participante deverá ter pleno conhecimento de si mesmo e dos outros agentes participantes.

Este conhecimento dos agentes participantes se dá através do compartilhamento de informações previstos no **CDM**. No CoDMAN os agentes são as aeronaves que negociam entre si a ocupação dos *slots*.

A Teoria dos Jogos propicia que uma parte das responsabilidades incidentes sobre os controladores de torre nos aeroportos seja transferida diretamente para os agentes racionais. Estes agirão em um cenário onde a força de negociação será o fator preponderante na efetiva alocação de *slots*, sendo cumpridas todas as exigências operacionais impostas por legislação e por restrições aeroportuárias. Tais demandas operacionais serão parte dos requisitos de arbitragem do mecanismo de negociação.

### 5.1.2 Modelagem do Cenário

O cenário foi modelado de acordo com os parâmetros operacionais do aeroporto e da área terminal:

- $F$ : conjunto dos voos ativos no aeroporto e na TMA. São voos ativos aqueles que ainda são alvo de processamento, discriminados em dois subconjuntos principais:
  - aeronaves em solo ( $F_g$ ), que inclui aquelas que já aterrissaram ( $F_g^-$ );
  - aeronaves no ar ( $F_a$ ), que inclui aquelas que já decolaram ( $F_a^-$ );
- $I_p$ : capacidade aeroportuária instalada, condicionada às posições de estacionamento ( $I_{PF}$ ) e às pistas de rolagem ( $I_{pr}$ );
- $C_a(t)$ : capacidade de aterrissagens do aeroporto no instante  $t$ ;
- $C_d(t)$ : capacidade de decolagens do aeroporto no instante  $t$ ;
- $Q_a$ : conjunto das aeronaves na TMA aguardando autorização para pouso;
- $Q_d$ : conjunto das aeronaves no pátio do aeroporto aguardando autorização para decolagem;
- $K$ : custo total dos atrasos;

### 5.1.3 Modelagem das Aeronaves

As aeronaves são definidas como agentes no cenário por isso é necessário que elas sejam niveladas e para que isso seja possível cada aeronave  $F_i$  recebe os seguintes atributos:

- $t_d$ : horário de decolagem previamente alocado para a aeronave com base no plano de voo;

- $t_a$ : horário previsto de aterrissagem no aeroporto de destino;
- $v$ : velocidade de cruzeiro;
- $t_f$  : duração estimada do voo até seu destino final;
- $\phi$ : fator de carga da companhia aérea, que é um mapeamento a um número real no intervalo  $[\frac{1}{2}, 1]$ ;
- $D$ : máximo período de atraso aceitável para a aeronave, em minutos;
- $p$ : número de passageiros presentes na aeronave;
- $B$ : porte da aeronave, mapeado em um número inteiro no intervalo  $[1, 3]$ ;
- $\sigma$ : significância da aeronave;
- $l$ : atraso imposto à aeronave, em minutos;
- $k$ : custo do atraso, que é a função de recompensa da aeronave;
- $S$ : *slot* alocado para a aeronave;

#### 5.1.4 Negociação de *Slots*

No início do processo as aeronaves são alocadas em uma fila de *slots* organizados na forma Primeiro a chegar, primeiro a sair (*First in, first out*) (FIFO). Cada aeronave  $F_i$  é inserida no sequenciamento e tem como objetivo anular  $k_i$ . Quando uma aeronave  $F_i$  deseja ocupar uma posição já ocupada por um outro voo, a aeronave a ser inserida força a manipulação desta fila, o que acarreta em atrasos a alguns ou todos os demais voos.

As aeronaves devem negociar umas com as outras de maneira colaborativa obedecendo ao CDM. A negociação visa minimizar o custo global dos atrasos. O custo total é dado pelas seguintes equações:

- Capacidade de decolagem:  

$$C_d(t) = I_p^r - (I_a(t) + I_d(t))$$
- Capacidade de aterrissagem:  

$$C_a(t) = \min\{I_p^f - F_g^-, I_p^r - (I_a(t) + I_d(t))\}$$
- Custo Total do Atraso:  

$$K = \sum_{i=1}^n k_i$$

As aeronaves podem assumir as seguintes posições durante a fase de negociação:

1. tentar capturar o *slot* seguinte, ganhando uma posição na fila;
2. ceder seu *slot* para outra aeronave, perdendo uma posição na fila;
3. permanecer em seu *slot*.

Sendo detectado o conflito de interesses, surge uma situação de negociação e o agente recebe um *payoff*. Este *payoff* não é diretamente utilizado como atributo de prioridade no sequenciamento, mas é útil para a mensurar a qualidade da negociação.

### 5.1.5 Avaliação dos Resultados

Os resultados exibidos na pesquisa são apresentados na Tabela 5.1 em que são confrontados valores de custo total e de seu desvio-padrão para alocação de *slots* de decolagem antes e após a realização da negociação segundo o protocolo de Rubinstein adaptado.

Ao se considerar o cenário padrão, em que a alocação dinâmica é simples, a otimização dos custos foi de 19,5%. O desvio-padrão dos custos foi o que obteve a maior redução, demonstrando que esta alocação é a que melhor distribui os custos dos atrasos pelas aeronaves.

A fase de negociação é importante pois permite às aeronaves uma oportunidade de ajustar seus *slots* de partida, observando seus interesses individuais. Esses interesses serão submetidos ao aeroporto que julgará se a oferta é ou não válida.

Cada aeronave será uma proponente e poderá propor uma oferta aceitável. Essa oferta tenderá a ser aquela que lhe traga maior benefício e a aeronave oponente acaba por aceitar tal oferta como plausível mesmo podendo não aceitá-la de início.

Caso a oferta de uma aeronave seja considerada inválida, ela perde a negociação, sendo aceita a última oferta válida da aeronave oponente. Caso contrário, a negociação entre as duas aeronaves não culminará num acordo e, assim, a entidade do aeroporto irá impor a decisão mais interessante para o cenário global.

Tabela 5.1: Comparação do desempenho dos resultados obtidos para a diminuição do custo total do cenário.

Cenário	Resultado com alocação padrão	Resultados após ofertas alternadas	Otimização
Sequenciamento estático	622.267 8.10712	518.097 7.26957	17,0% -0,83755
Sequenciamento dinâmico	733.767 8.93824	590.77 7.07508	19,5% -1,86316
Capacidade reduzida	2733.73 20.1882	2593.38 19.4701	5,1% -0,7181
Capacidade ampliada	605.067 8.0251	472.063 6.64978	23,0% -1,37532
Equalização PAX	554.5 6.8192	464.403 5.49119	16,2% -1,32801
Fator de carga uniforme	677.7 7.89842	544.55 6.9682	19,6% -0,93022

## 5.2 Gerenciamento de decolagem: Economizando o tempo de táxi, queima de combustível e emissões

O gerenciamento de decolagem possui como metas o aumento do *throughput* das *runways*, a redução do tamanho das filas, do tempo de táxi, queima de combustível e emissões. Uma ferramenta de Gerenciamento de Partidas (*Departure Management*) (DMAN) proposta pela *Sensis Corporation*[51] alcançou esses benefícios controlando o



momento em que cada aeronave realizava o *push back* do portão ou entrava na área de movimentação do aeroporto. Essa ferramenta automaticamente determina os horários de liberação para táxi e *take-off* de cada aeronave e permite aos seus usuários a modificação da agenda de voos de acordo com as necessidades.

### 5.2.1 Implementação e Objetivos

Quando a demanda supera a capacidade uma grande fila de decolagens é criada e o grande número de aeronaves nas *taxiways* inibe a habilidade dos controladores de tráfego aéreo de recriar uma sequência da fila visando um melhor *throughput* das *runways*. A ferramenta de DMAN visa diminuir essa ineficiência com:

1. Mantendo uma fila virtual de decolagens, no lugar de uma fila física na cabeceira da pista de decolagem.
2. Ajustando a sequência de decolagem para melhorar o *throughput* das *runways*.

Com a fila virtual as aeronaves se mantêm no portão ou em outro local com os motores desligados. Cada aeronave realiza a decolagem começando o táxi com tempo suficiente para se juntar a uma pequena fila física o que garante que as *runways* estejam sempre em uso, e permite ao controle de tráfego a oportunidade de melhorar a sequência de partida. O tamanho da fila física pode ser ajustado de acordo com as preferências do usuário, levando em consideração fatores como a configuração das *runways*, congestionamentos e condições meteorológicas.

Para melhorar o *throughput*, a ferramenta DMAN faz pequenos ajustes na sequência de partidas. Estes ajustes incluem agrupar pela classe das esteiras de turbulência e separar aeronaves que iram voar em trajetórias similares, entre outros. As alterações na sequência serão de apenas algumas posições na fila virtual, isso irá assegurar que nenhum voo sofra um grande atraso devido ao novo ressequenciamento. A sequência produzida pelo DMAN não impede que os controladores tenham a liberdade de escolher a sequência, eles podem ajusta-la de acordo com a necessidade.

### 5.2.2 Benefícios da ferramenta DMAN

Para a avaliação do estudo de caso foram feitas simulações de um dia inteiro em dois aeroportos: o Aeroporto Internacional John F. Kennedy (*John F. Kennedy International Airport*) (JFK) e o Aeroporto Internacional da Philadelphia (*Philadelphia International Airport*) (PHL), para cada aeroporto foram simulados dois cenários. O primeiro foi baseado no modelo *first-come-first-served*, que corresponde ao modelo em uso nos aeroportos. A acurácia do modelo foi validada através das gravações dos dados de monitoramento. O segundo cenário de simulação se baseia na utilização da ferramenta DMAN para segurar as aeronaves no portão e ajustar a sequência de partida.

A Tabela 5.2 mostra o tempo total de táxi na área de movimentação do aeroporto JFK. É mostrado o tempo total gasto de todas as aeronaves durante um dia, a média do tempo gasto em táxi por voo e o desvio padrão por tempo por voo. É possível observar que o DMAN economiza um total de 92 horas no tempo total na área de movimentação, ou 11 minutos por voo. No aeroporto PHL foram observados uma economia de 64 horas por dia ou 6 minutos por voo.

Tabela 5.2: Redução no tempo de táxi na área de movimentação do Aeroporto JFK.

	Sem DMAN	Com DMAN	Mudança
Total	197 hrs	105 hrs	92 hrs
Media	24 min	13 min	11 min
Desvio Padrão	15 min	4 min	11 min

Também foram comparados dados sobre o total de atraso através do tempo de espera na somado ao tempo de espera na área de movimentação como é observado na Tabela 5.3. Portanto, a redução em media de 1 minuto nos atrasos é equivalente ao tempo de melhora nas decolagens. No total, 11 horas de atrasos foram evitados pelo uso da ferramenta. Esta queda no tempo de atrasos é devido à melhorias no sequenciamento, que aprimora o *throughput* da *runway* de 1-2 aeronaves por hora. No aeroporto PHL os atrasos foram reduzidos em 2 minutos por decolagem, salvando um total de 24 horas por dia.

Tabela 5.3: Redução no tempo de atraso do Aeroporto JFK.

	Sem DMAN	Com DMAN	Mudança
Total	141 hrs	130 hrs	11 hrs
Media	17 min	16 min	1 min
Desvio Padrão	16 min	15 min	1 min

### 5.3 Realocação de recursos dos aeroportos em condições meteorológicas adversas.

Muitas vezes uma companhia aérea não pode utilizar seus *slots* devido à condições adversas, como, por exemplo, quando as condições meteorológicas não permitem o uso da pista de decolagem/aterrissagem. O tempo ruim pode diminuir a capacidade de aterrissagem nos aeroportos e, algumas vezes, até fechar o aeroporto, aumentando a demanda para aterrissagem em um horário posterior. Algumas estratégias para evitar que o aeroporto fique sobrecarregado são: diminuir a velocidade das aeronaves que estão à caminho, modificar as rotas e manter as aeronaves em pontos de retenção próximo ao aeroporto, porém todas estas estratégias acabam aumentando o consumo de combustível. Outra maneira de resolver este problema é fazendo com que as aeronaves permaneçam mais tempo no aeroporto de origem, forçando um atraso na chegada, esta técnica é conhecida como Programa de Atraso em Solo (*Ground Delay Program*) (GDP). Esta solução gera muitos *slots* não aproveitados.

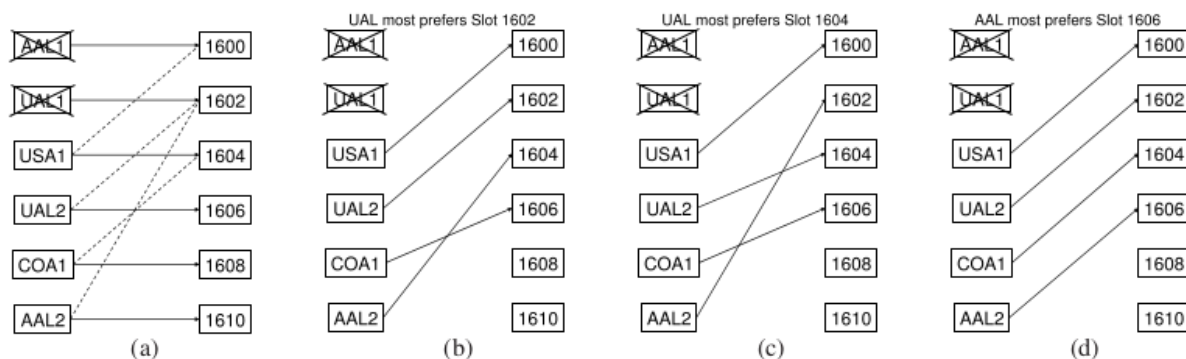
Uma solução proposta para este problema é a utilização do mecanismo de *Top Trading Cycles*[7], utilizado para alocação de casas, para fazer a alocação dos *slots*[7]. Os voos que passam pelo GDP recebem uma prioridade e são ordenados de acordo com ela. Se uma companhia aérea cancelar um voo e criou um *slot* vago, ela pode escolher um de seus voos como o voo de maior prioridade. Se um voo atrasar por razões que não envolvem o GDP, como, por exemplo, falhas mecânicas, a companhia aérea pode liberar seu *slot* e

usar sua prioridade para obter um *slot* em um horário posterior. Uma ordem é selecionada aleatoriamente para os voos restantes.

As companhias aéreas enviam suas preferências para cada voo. Na ausência de qualquer outra restrição, o conjunto de preferências definido por um voo  $f$  seria o conjunto de *slots* que vão do horário mais cedo possível de chegada,  $e(f)$  (o *slot* com maior preferência) até o *slot* atual,  $I(f)$ . O Algoritmo *Top Trading Cycles* é descrito a seguir:

1. Inicia-se com um conjunto de todos os voos e todos os *slots*.
2. Cada voo  $f$  aponta para seus *slots* preferidos, entre os *slots* restantes, de acordo com suas preferências definidas.
3. Cada *slot* ocupado aponta para seu ocupante e cada *slot* vazio aponta para o voo de maior prioridade entre os restantes.
4. O número de voos e *slots* é finito, então existe pelo menos um ciclo. Cada voo no ciclo recebe o *slot* para qual aponta e depois é removido.
5. Se existir pelo menos um voo e um *slot* restante, voltar para o passo 2.

Este algoritmo gera uma solução que é o ótimo de Pareto, ou seja, não existe nenhuma outra solução que seja preferida por todas as companhias aéreas, além da solução encontrada.



A Figura 5.1 ilustra a utilização do algoritmo. A Figura 5.1(a) mostra a alocação dos *slots* após o RBS. As linhas sólidas representam as atribuições iniciais de *slots* para cada voo e as linhas pontilhadas mostram o tempo de chegada mais cedo possível. As Figuras 5.1(b), 5.1(c) e 5.1(d) mostram como a distribuição dos *slots* seria modificada dependendo da preferência de cada companhia aérea. A Figura 5.1(b) mostra a alocação de *slots* se a empresa UAL preferir o *slot* 1602, a Figura 5.1(c) mostra a alocação se a UAL preferir o *slot* 1604 e a Figura 5.1(d) mostra a alocação se a empresa AAL preferir o *slot* 1606.

## 5.4 SIVOR - *Simulateur de VOI en Route*

O Simulador de voo em rota (*Simulateur de VOI en Route*) (SIVOR)[21] é um protótipo criado para simular o ATFM. Neste sistema, quando uma aeronave não consegue utilizar

seu *slot* alocado, ocorre um evento chamado de *alea*. Para acomodar os *aleas*, criaram a idéia de Área de Absorção (AA). As AAs são um número de *slots* que são mantidos vazios durante o processo de alocação de *slots*, permitindo a absorção de perturbações com a menor modificação possível no planejamento original. Encontrar a melhor configuração de AAs corresponde a alcançar dois objetivos: maximizar seus efeitos e minimizar os *slots* perdidos.

Seja  $n$  o número total de *slots* e  $v_1 = n$  a vazão no caso sem AAs e que todos os *slots* são utilizados de acordo com o planejamento inicial. Este caso é o ideal, porém nunca acontece na realidade, algumas aeronaves não utilizam seus *slots* e precisam ser realocadas. A vazão total vai ser sempre menor que  $v_1$ .

Seja  $p$  a probabilidade de que uma aeronave utilize seu *slot*. A vazão no caso em que não se utiliza as AAs é dada por  $v_2 = pn$ . A vazão com a utilização das AAs é dada por  $v_3 = p(n - z) + \min\{z, (1 - p)(n - z)\}$ , onde  $z$  é o número de *slots* não utilizados. Para  $v_3$  são alocadas  $(n - z)$  aeronaves. O total de aeronaves decolando em seus *slots* é dado por  $p(n - z)$ . É necessário encontrar o melhor valor de  $z$  para obter o melhor  $v_3$ . A maior capacidade é obtida quando todas as aeronaves que obtêm *aleas* conseguem *slots* vazios. Mais precisamente:

$$z = (1 - p)(n - z) \quad (5.1)$$

$$n - z = n - n \frac{1 - p}{2 - p} \quad (5.2)$$

$$z = n \frac{1 - p}{2 - p} \quad (5.3)$$

$$v_3 = \frac{n}{1 + (1 - p)} \quad (5.4)$$

Para mostrar a vantagem de se utilizar as AAs, é necessário que  $v_3 - v_2 > 0$ , e, de fato, isto ocorre:

$$v_3 - v_2 = n - z - pn \quad (5.5)$$

$$v_3 - v_2 = n \left(1 - \frac{1 - p}{2 - p} - p\right) \quad (5.6)$$

$$v_3 - v_2 = n \frac{(p - 1)^2}{2 - p} \quad (5.7)$$

Esta diferença é sempre positiva para todo  $p \in [0, 1[$ . Sendo assim, fica provado que utilizar AAs gera uma melhor vazão do que não utilizá-las.

# Capítulo 6

## Sequenciamento de Partidas de Aeronaves

Este capítulo apresenta o problema de sequenciamento de partidas, na Seção 6.1 é apresentada a definição do problema de sequenciamento de partidas, a Seção 6.2 apresenta informações sobre o Aeroporto Internacional de Brasília, a Seção 6.3 expõe como é realizado o sequenciamento de partidas no Aeroporto Internacional de Brasília e quais os problemas encontrados. A forma como os dados do aeroporto são coletados é apresentada na Seção 6.4 e a influência de atrasos e cancelamentos no sequenciamento de partidas é mostrada na Seção 6.5.

### 6.1 Problema

O foco de estudo é a coordenação do processo de sequenciamento de partidas de aeronaves, que no contexto operacional brasileiro, é dependente exclusivamente da experiência do controlador, que, muitas vezes empiricamente, decide a alocação dos *slots* de partida das aeronaves.

A alocação empírica de *slots* apesar de funcional, não é a ideal, pois ela não respeita o planejamento prévio e atende sempre o primeiro piloto a solicitar o *push-back*, que é a solicitação para a aeronave iniciar o processo de partida, e isso pode gerar filas pois podem haver mais solicitações que o tempo gasto entre a *taxiway* e a *runway*, causando uma fila física de aeronaves que consomem mais combustível, aumentam a poluição e levam mais tempo no processo de decolagem.

A proposta de pesquisa veio a fim de tratar o problema do gerenciamento de partidas em aeroportos. É proposta uma abordagem experimental de gerenciamento colaborativo de partidas, onde cada agente no modelo é uma aeronave a impactar o cenário das movimentações aéreas no aeroporto de análise.

Este trabalho foca no sequenciamento de partida no Aeroporto Internacional de Brasília — Presidente Juscelino Kubitschek.

## 6.2 Aeroporto Internacional de Brasília

No dia 23 de outubro de 2013, os autores deste trabalho tiveram a oportunidade de visitar o Centro Operacional do Aeroporto de Brasília e obter as informações contidas neste capítulo.

O Aeroporto Internacional de Brasília é o terceiro aeroporto mais movimentado do Brasil, em termos de passageiros e aeronaves, ficando atrás apenas do Aeroporto de Congonhas e do Aeroporto de Guarulhos. Com as obras que estão sendo realizadas em 2013, é previsto que ele ultrapasse o Aeroporto de Congonhas neste *ranking*. São mais de 16 milhões de passageiros por ano e possui uma capacidade de até 40 aeronaves comerciais no pátio, a Tabela 6.1 exibe os dados de número de passageiros e número de aeronaves que circularam pelo aeroporto em outubro de 2013.

Tabela 6.1: Dados do Aeroporto Internacional de Brasília em outubro de 2013 [30].

Dados	Doméstico	Internacional	Total
Aeronaves(unid.)	14.946	367	15.313
Passageiros (unid.)	1.418.065	45.771	1.463.836

Estes dados mostram a importância do Aeroporto Internacional de Brasília no cenário nacional e internacional, fazendo dele um ótimo estudo de caso, que poderia ser aplicado a outros aeroportos, também de grande porte.

A Figura 6.1 mostra o diagrama do Aeroporto Internacional de Brasília, é possível perceber que ele possui duas pistas de decolagem ou aterrissagem. Geralmente uma pista é utilizada para a decolagem e a outra é utilizada para aterrissagem, porém podem ocorrer situações em que se precise utilizar ambas para aterrissagem ou decolagem, como, por exemplo, quando ocorre fechamento de outros aeroportos internacionais e várias aeronaves precisam ser redirecionadas para o Aeroporto Internacional de Brasília.

## 6.3 Sequenciamento de Partidas no Aeroporto de Brasília

O Aeroporto de Brasília não utiliza um sistema de *slots*, os horários de decolagem dos voos são definidos de acordo com o Horário de Transporte (Hotran). O Hotran é o documento aprovado e emitido pela ANAC que formaliza as concessões para a exploração de linhas aéreas regulares internacionais e domésticas de passageiros e/ou carga e da rede postal pelas empresas de transporte aéreo, com os respectivos horários, números de voos, frequências, tipos de aeronaves e oferta de assentos[1].

Cada voo terá um horário de chegada e partida previsto pelo Hotran, porém é muito comum este horário previsto não corresponder com a realidade. Frequentemente ocorrem atrasos, principalmente devido à chuva e outras condições meteorológicas. O CGNA é que controla o tempo real de chegada ou partida, liberando a aeronave para pouso ou decolagem, dependendo das condições da pista. Na ocorrência de atraso, se a pista

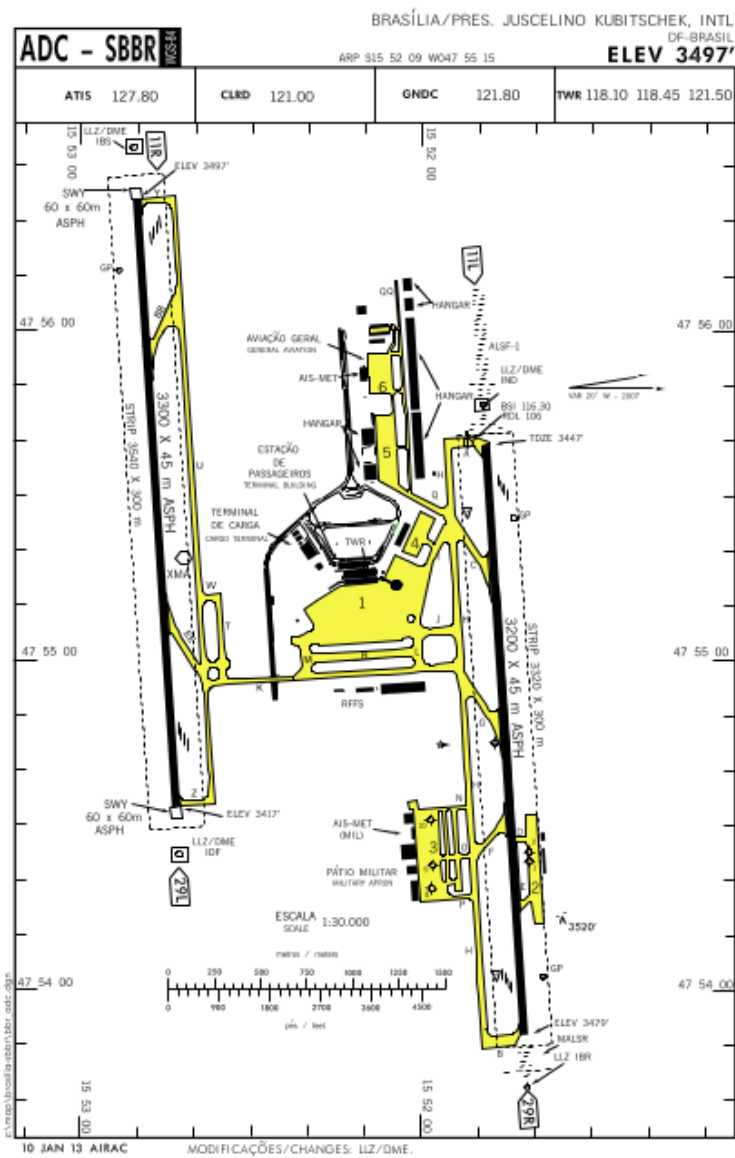


Figura 6.1: Diagrama do Aeroporto Internacional de Brasília [4].

de decolagem/aterriagem estiver livre, a primeira aeronave que realizar o pedido para utilizar a pista, será atendida (*first-come first-served*). Da mesma forma, se um voo for cancelado, qualquer outra aeronave pode realizar o pedido para utilizar a pista em seu lugar.

Este modelo possui a vantagem de ser simples e não exige uma grande organização por parte dos controladores de voo, é de fácil implementação e, como foi visto no histórico da alocação de *slots*, foi uma das primeiras formas de se utilizar as pistas de decolagem/aterriagem. Porém, esta simplicidade nem sempre garante eficiência em termos de atrasos, uso de combustível, número de passageiros e outros fatores.

Este modelo possui o problema de não permitir uma otimização planejada do uso da pista, pois o controle aéreo deve ficar à espera da resposta das aeronaves para ter o conhecimento de qual será a próxima a utilizar a pista. O modelo não permite que os controladores de voo tenham um controle maior das atividades na pista. Além disso, muitas vezes se formam longas filas para utilizar as pistas, fazendo com que as aeronaves fiquem paradas com o motor ligado, gastando mais combustível e gerando uma emissão desnecessária de poluentes.

## 6.4 Coleta de Dados

Os dados para teste foram obtidos do site da Infraero, coletados com um *crawler* construído para capturar as informações dos voos em tempo real. A Tabela 6.2 mostra um exemplo de alguns dados coletados das partidas do Aeroporto de Brasília, na primeira coluna estão as empresas aéreas e seus voos na segunda coluna, na terceira estão os destinos dos voos, na quarta coluna estão os horários previstos dos voos, que é o horário agendado para a saída do voo e na quinta coluna estão os horários confirmados, que é o horário real em que a aeronave decolou. Na sexta coluna estão os status dos voos e na sétima coluna estão os atrasos de cada voo, este atraso foi calculado pela diferença do tempo confirmado e o tempo previsto, se esta diferença for negativa significa que o voo foi adiantado e não há atraso, como no caso do voo 4427.

Tabela 6.2: Dados de partida dos voos obtidos do site da Infraero [29].

Empresa	Voo	Destino	Previsto	Confirmado	Status	Atraso
TAM	3739	São Paulo - SP	08:01	08:06	Finalizado	00:05
GOL	1683	Guarulhos - GRU	08:15	08:20	Finalizado	00:05
AZUL	4427	Campinas - SP	08:18	08:13	Finalizado	00:00
GOL	1081	Belo Horizonte - MG	08:23	08:44	Finalizado	00:21
GOL	1403	São Paulo - SP	08:29	08:21	Finalizado	00:00
GOL	1144	Campinas - SP	08:31	08:42	Finalizado	00:11

## 6.5 Atrasos e Cancelamentos

Existem muitas razões para atrasos e cancelamentos de voos, as principais são [49]:

- Condições meteorológicas adversas;



- Falhas humanas ou mecânicas imprevistas;
- Aeroporto congestionado;
- Espaço aéreo congestionado;

O sistema de sequenciamento de partidas do Aeroporto Internacional de Brasília somente cria uma agenda com os voos e seus Hotrans, ele não consegue se adaptar às situações em que ocorrem atrasos ou cancelamentos. Quando ocorre um atraso, geralmente este atraso gera um efeito dominó, que faz com que todos os outros voos atrasem também, até que a quantidade de voos diminua e o aeroporto fique menos sobrecarregado.

Para diminuir os prejuízos econômicos e ambientais dos atrasos ocorridos em solo, ou seja, quando a aeronave atrasa seu horário de partida é fundamental que [5]:

1. A eficiência do uso da pista seja aumentada;
2. Os atrasos nas filas para uso da pista sejam diminuídos;
3. O tempo de táxi para a pista seja diminuído;
4. O tempo em que o motor fica ligado seja minimizado;

O uso de um sistema de gerenciamento de partidas eficiente torna possível alcançar os itens 1, 2 e 4. O item 3 já foi tratado em outra pesquisa do TransLab sobre gerenciamento de *taxiways* [39].

Pesquisas que foram financiadas pela Administração Nacional da Aeronáutica e do Espaço (*National Aeronautics and Space Administration*) (NASA) sobre a causa de atrasos, chegaram à conclusão que ferramentas automatizadas para auxiliar a otimização do sequenciamento de partidas é benéfico tanto para os controladores de voo, quanto para os operadores de aeronaves [5].

# Capítulo 7

## Modelagem e Implementação

O modelo deste trabalho tem como objetivo satisfazer duas motivações:

1. Diminuir o atraso total dos voos, em situações de congestionamento, comparando com o atraso real.
2. Beneficiar as empresas aéreas com *slots* melhores, caso elas notifiquem atrasos e cancelamentos com antecedência.

Neste modelo será analisada apenas a decolagem, supondo que o Aeroporto Internacional de Brasília utiliza uma pista exclusiva para decolagem e outra para aterrissagem.

Para alcançar estes objetivos são aplicados os algoritmos genéticos à alocação de *slots*, baseando-se nos algoritmos de RBS e *Compression*.

A utilização de *slots* beneficiaria as empresas aéreas, permitindo que elas saibam com antecedência seus horários confirmados de voo e também beneficiaria os controladores de voo, diminuindo a carga de decisões que eles suportam.

Este capítulo apresenta como foi realizada a modelagem e a implementação da proposta, a Seção 7.1 apresenta como é realizada a alocação inicial dos *slots*, a Seção 7.2 apresenta como foi realizada a modelagem do algoritmo genético e de todas suas estruturas, na Seção 7.3 as restrições para onde cada voo pode ser alocado são mostradas, por fim, na Seção 7.4 detalhes da implementação são mostrados.

### 7.1 Alocando *slots*

Conforme foi mencionado na Seção 6.3, o Aeroporto de Brasília não utiliza *slots*, para organizar o sistema de acordo com a solução proposta é necessário que os voos sejam divididos em *slots*. Para utilizar *slots* deve-se, primeiramente, definir o tempo de cada *slot*, o tamanho de *slot* geralmente utilizado pela FAA é de 1 minuto. Em situação de chuva ou outros eventos meteorológicos o tamanho do *slot* pode ser aumentado para 2 minutos. O tamanho do *slot* também pode ser calculado a partir da capacidade do aeroporto.

A Figura 7.1 apresenta como seria realizada a alocação dos *slots*, caso os *slots* possuíssem 5 minutos de tamanho. Caso dois ou mais voos caíssem no mesmo *slot*, o voo com menor horário ficaria com o *slot* e os outros voos seriam realocados para os próximos *slots*, conforme é mostrado na Figura 7.2

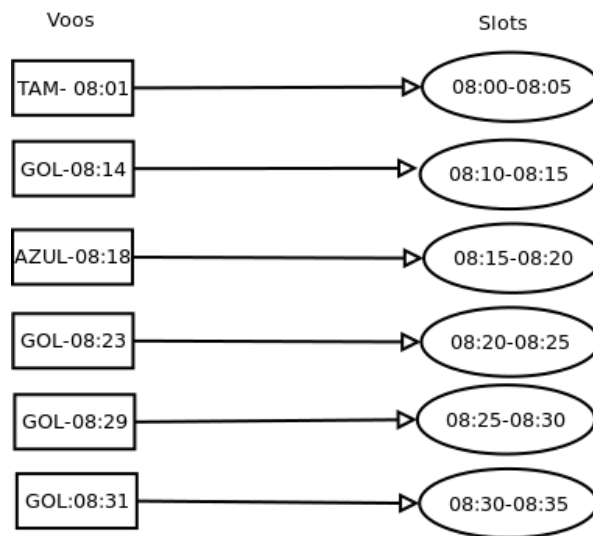


Figura 7.1: Alocação dos *slots*.

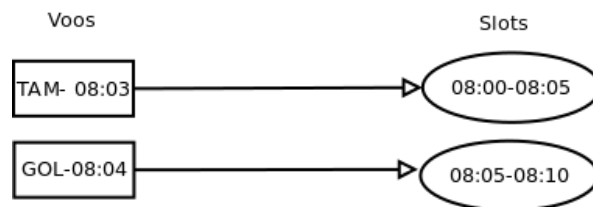


Figura 7.2: Alocação dos *slots* em horários próximos.

## 7.2 Modelagem do Algoritmo Genético

Um algoritmo genético típico precisa de:

1. Uma representação genética do domínio de solução.
2. Uma função *fitness* para avaliar este domínio de solução.

### 7.2.1 Representação genética

Esta seção apresenta a representação genética do problema, considerando cada uma das seguintes estruturas do algoritmo genético: gene, cromossomo, população, seleção, mutação e *crossover*.

#### Gene

Nesta solução, os voos formam os genes e cada *slot* será um *locus* (a posição do gene). Ou seja, as sequências de voos são os indivíduos da população. Cada voo só poderá pertencer a um *slot*, senão a aeronave teria dois horários diferentes para decolar, o que não é possível.

Cada voo é representado por seu número e pela companhia aérea à que pertence, é um identificador único de seis dígitos, sendo os dois primeiros um identificador que representa

a companhia aérea(Tabela 7.1) e os outros quatro dígitos é o número do voo. A Figura 7.3 ilustra este modelo, onde tem-se 6 *slots* e 6 voos diferentes.

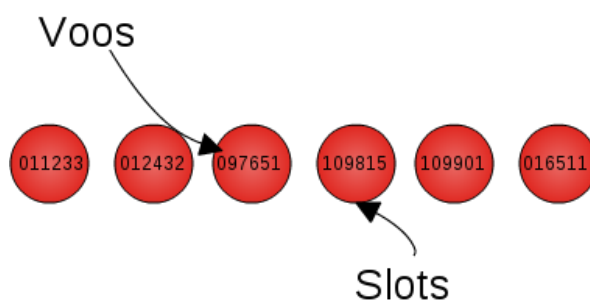


Figura 7.3: Estrutura dos genes e do cromossomo.

Tabela 7.1: Códigos para algumas das empresas aérea.

Companhia	Codificação
GOL	0
TAM	1
AZUL	2
AVIANCA - ONE	3
RIO LINHAS AÉREAS	4
PASSAREDO	5
SETE	6
TRIP	7

## Cromossomo

Cada cromossomo representa uma possível solução do problema, ele é composto por vários genes. A Figura 7.3 mostra um cromossomo composto por 6 entradas. Nesta solução, o cromossomo é formado por um *array* de números, onde cada elemento do *array* é um voo que pertence a uma empresa aérea. O intervalo trabalhado será de quatro horas, pois é a duração do horário de pico do aeroporto, como cada *slot* possui um minuto de duração, cada cromossomo será formado por 240 *slots*.

## População

A população é formada pelo conjunto de cromossomos, neste caso, é formada por todas as sequências de saídas geradas através dos mecanismos de reprodução e possíveis soluções.

## Seleção, Mutação e *Crossover*

A mutação ocorrerá com uma probabilidade  $P_m$  entre 0,01 e 0,005, esta probabilidade deve ser baixa pois uma taxa alta de mutação faz com que o algoritmo expanda seu espaço de busca ainda mais, tornando a convergência mais demorada [21]. A mutação será realizada trocando-se dois *slots* de um cromossomo de forma aleatória. O *crossover* realizado será do tipo uniforme baseado em ordem (Figura 2.4) devido ao fato de que neste modelo não é permitido que existam genes com valores repetidos, cada voo só pode pertencer a um *slot*. As probabilidades de *crossover* testadas neste trabalho são  $P_C = 0.7$  e  $P_C = 0.9$ , pois são as probabilidades recomendadas [36].

O Método da Roleta será utilizado para selecionar os indivíduos da população, sua vantagem é que ele possibilita uma chance maior de seleção dos indivíduos com melhor valor de *fitness*, porém não descarta os demais.

A cada nova geração, 20% dos indivíduos com maior valor de *fitness* serão mantidos na geração seguinte, junto com os novos indivíduos, preservando as melhores soluções.

O algoritmo será executado até que ocorra a convergência e a função *fitness* pare de variar ao longo do tempo, em um determinado número de gerações, chegando a uma solução ótima.

### 7.2.2 Função *fitness*

O modelo tem como objetivos principais diminuir o tempo de atraso e beneficiar empresas que avisem atrasos/cancelamentos com antecedência. A função *fitness* deve refletir estes objetivos, avaliando todas as soluções e verificando quais se aproximam mais de uma solução desejável.

$$f = \begin{cases} \frac{1}{(\sum_{j=1}^n t_{atraso_j})} \times e, & \text{se } e > 0 \\ \frac{1}{(\sum_{j=1}^n t_{atraso_j})}, & \text{se } e = 0 \end{cases} \quad (7.1)$$

$$t_{atraso} = \begin{cases} t_{pc_j} - t_{pp_j}, & \text{se } t_{atraso} \geq -5 \\ |t_{pc_j} - t_{pp_j}|, & \text{se } t_{atraso} < -5 \end{cases} \quad (7.2)$$

A Equação 7.1 mostra a função *fitness*, onde  $n$  é o número total de voos no cromossomo e  $t_{atraso_j}$  é o tempo de atraso do voo  $j$ , este tempo é dado pela equação 7.2. A variável  $e$  representa o número de empresas aéreas que liberaram um *slot* e foram beneficiadas, cada empresa beneficiada gera um aumento na função *fitness*. Se nenhuma empresa for beneficiada, é considerado somente o tempo de atraso. Para esta solução, quanto maior o valor de *fitness*, mais próximo o sistema estará da solução ideal.

A Equação 7.2 mostra o como é calculado o tempo de atraso, onde  $t_{pc_j}$  é o tempo de partida confirmado para o voo  $j$  e  $t_{pp_j}$  é o tempo de partida previsto para o voo  $j$ . Neste trabalho é suposto que cada voo pode ser adiantado em no máximo 5 minutos, logo, caso o voo seja adiantado em até 5 minutos, este valor continuará negativo para diminuir o tempo total de atraso e melhorar a função *fitness*, porém caso o valor seja menor que -5 o voo já estará adiantado demais, não podendo decolar neste horário, então o valor total de atraso deverá ser aumentado para forçar uma diminuição na função *fitness*.

### 7.3 Restrições na Alocação de *slots*

Algumas restrições devem ser impostas na alocação de *slots*, pois existem algumas soluções que não poderão ser adotadas. Nesta seção, estas restrições são apresentadas.

As companhias aéreas que avisarem o cancelamento com antecedência, terão prioridades na utilização do *slot* que elas liberaram, conforme é realizado no RBS e no *Compression*. Porém, existem casos em que o *slot* de outro voo mais próximo da companhia aérea está muito distante para utilizar o *slot* liberado, então devem-se alocar os voos mais próximos de outras companhias e então alocar um outro *slot* melhor para o voo da companhia aérea que informou o cancelamento. A Figura 7.4 mostra que o voo 5091 da

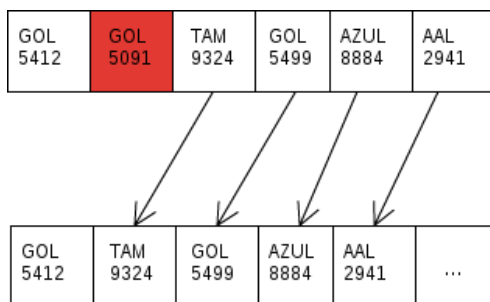


Figura 7.4: O voo da companhia aérea que liberou o *slot* não pode utilizar este *slot*, então outro voo foi realocado para melhorar o *slot* de outro voo da GOL.

GOL foi cancelado, o ideal era que outro voo da mesma companhia aérea ocupasse este *slot*, porém, neste caso, o voo não pode ocupá-lo, então o voo 9324 da TAM foi alocado para este *slot* e o voo da GOL pode utilizar um *slot* melhor.

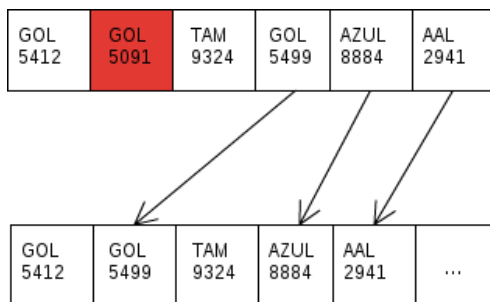


Figura 7.5: O voo da companhia aérea que liberou o *slot* pode utilizar o *slot* liberado.

A Figura 7.5 ilustra um caso onde o voo da GOL pode ser realocado para o *slot* liberado e a Figura 7.6 mostra um caso onde nenhum voo pode utilizar aquele *slot*, isto pode ocorrer quando o *slot* cancelado estiver muito distante dos outros. Baseando-se na média dos voos coletados da Infraero, um voo pode ser adiantado em no máximo 5 minutos de seu horário previsto. Também pode ocorrer o caso em que a pista não pode ser utilizada por questões meteorológicas.

GOL 5412	GOL 5091	TAM 9324	GOL 5499	AZUL 8884	AAL 2941
GOL 5412	X	TAM 9324	GOL 5499	AZUL 8884	AAL 2941

Figura 7.6: Nenhum voo pode utilizar o *slot* liberado.

## 7.4 Implementação da proposta

A Figura 7.7 apresenta a modelagem do sistema implementado. O sistema foi implementado em linguagem C devido ao fato de permitir uma abordagem simples e direta para o problema e uma maior eficiência por ser uma linguagem procedural.

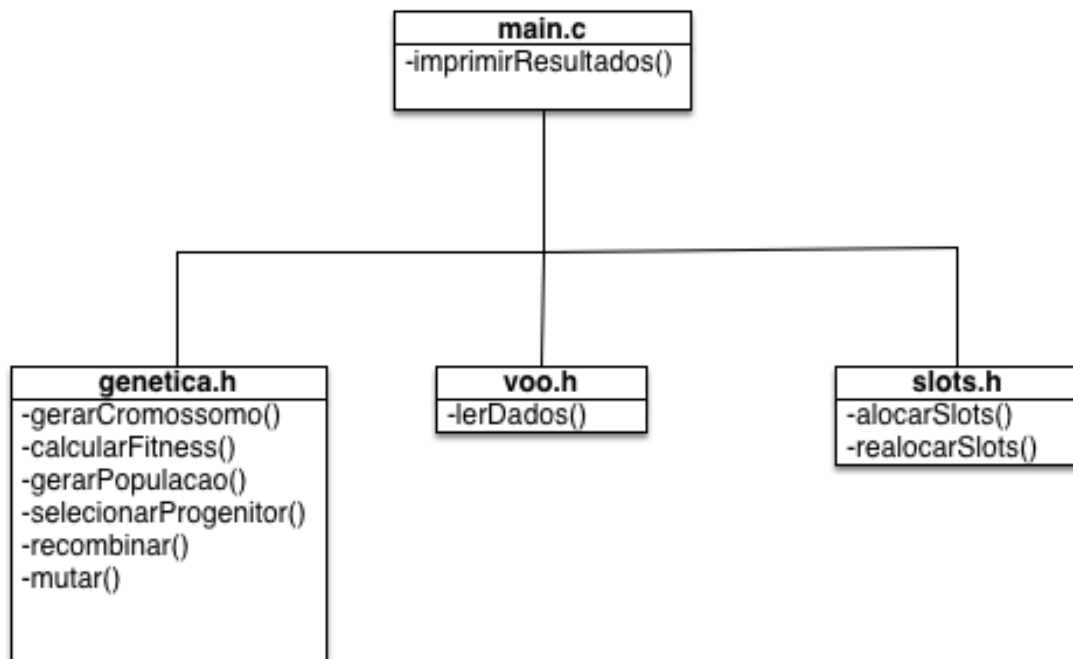


Figura 7.7: Diagrama representando a implementação.

A implementação foi dividida em três módulos principais, **genetica.h**, **voo.h** e **slots.h**.

### Módulo de Voo

Foi implementado em **voo.h**, este módulo é responsável pela leitura dos dados das partidas dos voos através da função **leDados()**, ele também contém a estrutura para armazenar cada um destes voos.

## Módulo de *Slots*

Implementado em **slots.h**, os *slots* são armazenados em uma estrutura contida neste módulo, a função **alocarSlots()** realiza a alocação inicial dos *slots*, como o Aeroporto Internacional de Brasília não utiliza *slots* é necessário que cada voo seja designado para o *slot* mais próximo de seu horário de partida. A função **realocarSlots()** realiza uma nova alocação de *slots* utilizando algoritmos genéticos, esta realocação de *slots* leva em consideração os *slots* vazios e os *slots* liberados devido a atrasos e cancelamentos.

## Módulo de Genética

Este módulo é responsável por todas as operações relacionadas aos algoritmos genéticos, ele foi implementado em **genetica.h**. A função **gerarCromossomo()** gera um cromossomo com uma distribuição aleatória de *slots* para cada voo, com exceção dos voos que foram marcados como fixos (os atrasos e cancelamentos avisados). A função **calcularFitness()** calcula o valor de *fitness* para cada cromossomo utilizando a equação 7.1.

A função **gerarPopulacao()** chama a função **gerarCromossomo()**  $n$  vezes, sendo  $n$  o tamanho da população determinado. A **selecionaProgenitor()** seleciona dois progenitores para a reprodução utilizando o método da roleta, a chance de cada cromossomo ser escolhido é proporcional a seu valor de *fitness*.

O *crossover* é realizado através da função **recombinar()**, dada uma taxa de *crossover*, ele irá gerar uma sequência aleatória de números binários, a probabilidade do número ser 1 nesta sequência é igual a taxa de *crossover* e o *crossover* irá ocorrer nos pontos em que a *string* binária é igual a 1.

A mutação ocorre de acordo com uma taxa de mutação determinada, ela é realizada pela função **mutar()**. A mutação é realizada trocando-se a posição de dois *slots* aleatoriamente de acordo com a taxa de mutação.

Existe ainda a função **main()** em um módulo separado, que irá simplesmente chamar a execução dos outros módulos e imprimir o resultado das alocações realizadas.

A Figura 7.8 apresenta o fluxograma desta implementação. No início são lidos os dados de um arquivo que contém as informações dos voos, depois é realizada a alocação inicial dos *slots*, que serve de entrada para o algoritmo genético. O algoritmo genético gera várias soluções e a melhor delas é impressa e o programa encerra.

A Figura 7.9 apresenta como é feita a parte de alocação de *slots*, inicialmente os *slots* são criados com um tamanho determinado, neste caso 1 minuto, e então inicia-se a alocação dos *slots*. Os voos fixos, cujos atrasos foram notificados pelas empresas aéreas e não podem ser modificados são inseridos nos *slots* de seus horários confirmados. Os outros voos são alocados em um *slot* aleatório, esta sequência é passada para a etapa do algoritmo genético.

A Figura 7.10 apresenta como foi estruturado o algoritmo genético, primeiro é gerado um cromossomo a partir da sequência de *slots* de entrada, os cromossomos são gerados até que a população seja grande o suficiente. O valor de *fitness* de cada indivíduo é então avaliado. Dois indivíduos são selecionados pelo método da roleta para reproduzirem e criarem uma nova geração através do *crossover*, gerando dois filhos, os indivíduos gerados passam por uma mutação com uma probabilidade definida. Este processo segue até gerar a quantidade necessária de filhos. Os melhores pais são selecionados para continuar na



próxima geração e o restante é deletado. Continua criando gerações até atingir o número determinado e então envia a melhor solução para que possa ser impressa.

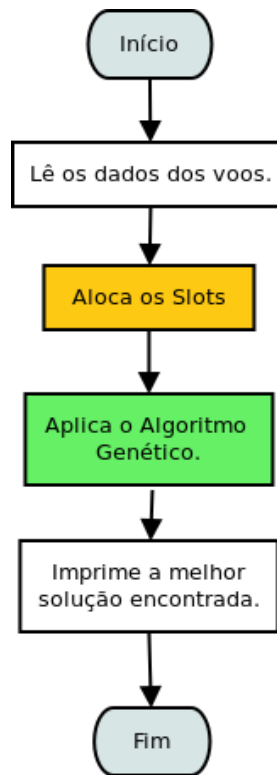


Figura 7.8: Fluxograma da implementação proposta.

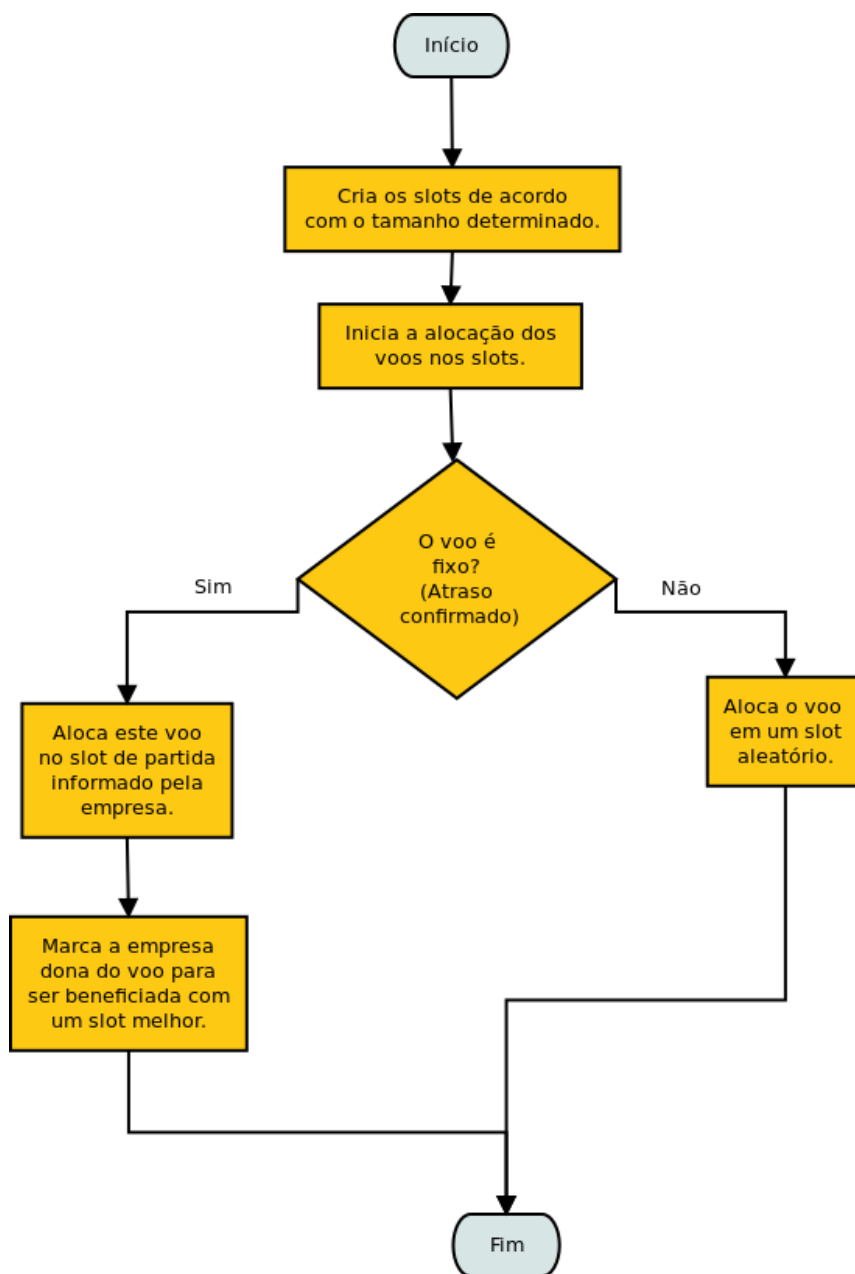


Figura 7.9: Fluxograma da alocação de slots.

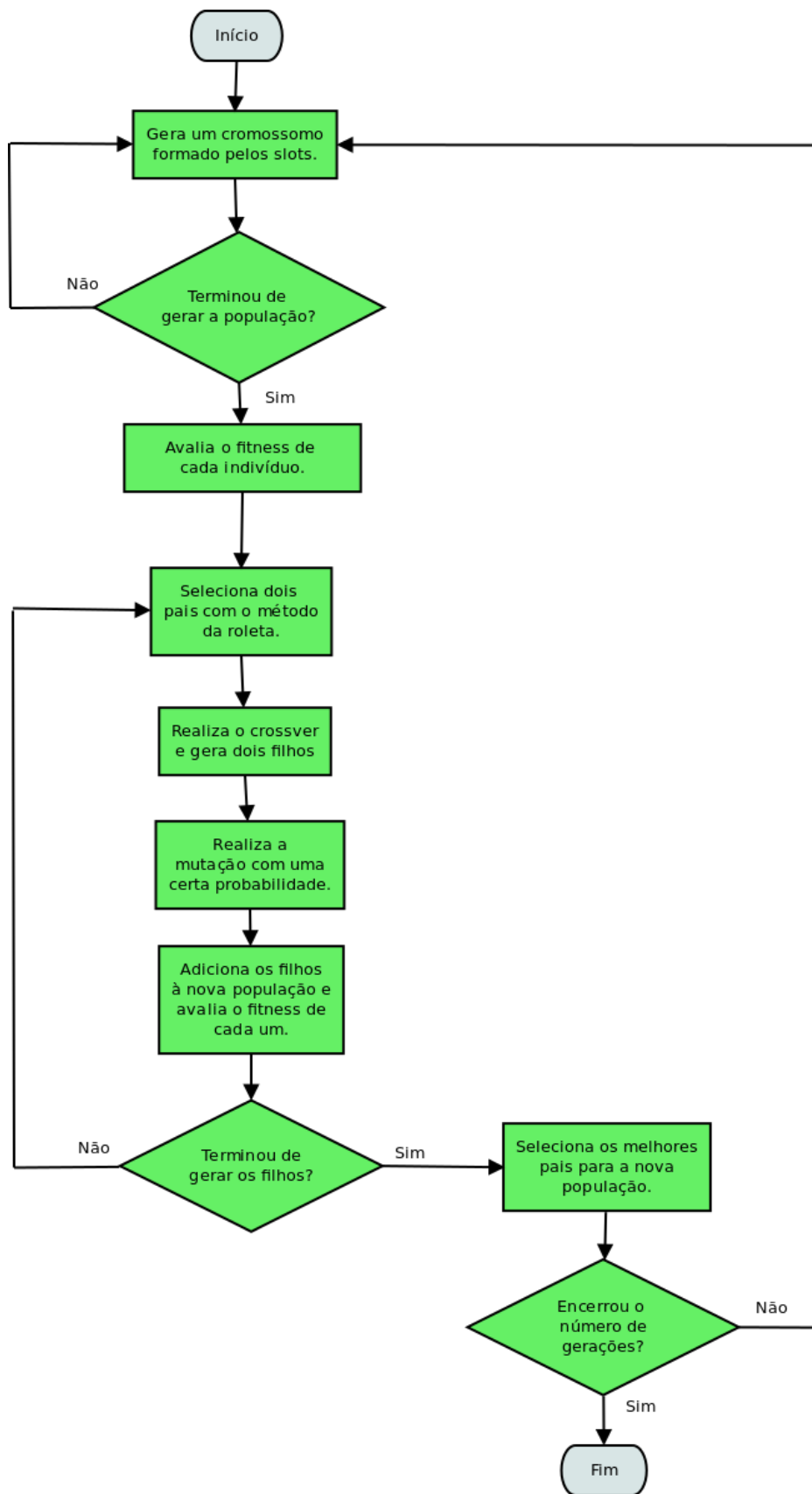


Figura 7.10: Fluxograma do algoritmo genético.

# Capítulo 8

## Estudo de Caso

Neste Capítulo é apresentado o estudo de caso utilizado para testar a metodologia implementada no Capítulo 7, para realizar os testes foi utilizado o ambiente apresentado na 8.1, na seção 8.2 é apresentado como foram planejados os casos de testes e nas seções 8.3 e 8.4 são apresentados os dois cenários estudados. Na seção 8.5 são apresentados os resultados obtidos e nas seções 8.6, 8.7 e 8.8 são analisados os resultados com relação aos dados reais e ao CoDMAN[53].

### 8.1 Ambiente de Teste

O ambiente foram realizados os testes do modelo é detalhado na Tabela 8.1:

Tabela 8.1: Ambiente de testes

Hardware	Intel(R) Core(TM) i3 CPU M 370 @ 2.40GHz x 4, 2,8 GiB
Sistema Operacional	ubuntu 12.04 LTS
Compilador	gcc version 4.6.3
IDE	KATE Development Platform: 4.8.5 (4.8.5)

### 8.2 Planejamento de Casos de Testes

Foram planejados dois cenários para teste, um durante o horário de pico e outro fora do horário de pico. Este trabalho tem como objetivo diminuir os atrasos das decolagens durante os períodos de congestionamento nos aeroportos, por isso é fundamental verificar a eficiência para os horários de pico. Mas é também interessante verificar como o sistema se comporta em situações em que os atrasos não são tão significativos.

Foi escolhida a mesma data para ambos cenários, para evitar que a diferença entre os dias da semana influenciasse no resultado. Também é importante notar que os dados foram coletados em um dia ensolarado, logo, as condições meteorológicas não influenciaram no resultado.

Para verificar a sensibilidade do modelo foram variados dois parâmetros: a taxa de mutação e a taxa de *crossover*. A taxa de mutação será variada em 0,005 e 0,01 e a taxa

de *crossover* em 0,7 e 0,9. Também foi necessário testar e identificar qual o tamanho ideal para a população e o número de gerações em que ocorria a convergência.

### 8.3 Cenário 1: Horário de pico

De acordo com o Centro Operacional do Aeroporto Internacional de Brasília, o horário de pico, quando há mais atraso e maior movimento, é entre oito da manhã e meio dia. Este cenário engloba os voos comerciais que partiram do aeroporto dentro deste horário de pico, os dados coletados foram do período das 8:00 até 12:00 do dia 20 de novembro de 2013. As informações resumidas deste cenário são:

- A soma total dos atrasos deste período foi de 21 horas e 37 minutos.
- A média de atraso por voo foi de 26 minutos.
- Foi realizado um total de 49 decolagens.

### 8.4 Cenário 2: Horário normal

Neste cenário foi escolhido um período de tempo fora do horário de pico. Para que os resultados deste cenário fossem comparáveis ao cenário 1, foi necessário escolher um período de tempo equivalente ao tamanho do período do cenário 1, um período de quatro horas. O período escolhido para coletar os dados foi das 16:00 até 20:00 do dia 20 de novembro de 2013. As informações resumidas deste cenário são:

- O atraso total deste período foi de 11 horas e 12 minutos.
- A média de atraso por voo foi de 16 minutos.
- Foram realizadas 43 decolagens.

## 8.5 Resultados Obtidos a partir da Proposta

Os dados dos voos obtidos do site da Infraero não contém as circunstâncias dos atrasos, para que seja possível aplicar este algoritmo foi suposto que os voos com atraso maior que 30 minutos não poderiam ter este atraso diminuído e que as companhias aéreas informaram este atraso, sendo recompensada com *slots* melhores.

### 8.5.1 Tamanho da População

Não existe um tamanho padrão para a população, mas é importante que ela não seja pequena demais, evitando que a solução ótima seja encontrada e não pode ser grande demais, para que o tempo de convergência seja razoável. Para esta solução, o tamanho da população  $n = 1000$  apresenta um bom tempo de execução e uma melhor solução em comparação com os dados reais, como será observado nos resultados.

## 8.5.2 Número de gerações

O número de gerações necessário para que uma população de tamanho  $n = 1000$  atinja convergência é aproximadamente igual a 100, conforme mostra o gráfico da Figura 8.1. A Figura 8.1 exibe que depois de 100 gerações, o valor de *fitness* deixa de variar significativamente em ambos os casos de estudo (no horário de pico e fora do horário de pico). Observe que o valor de *fitness* será sempre um número pequeno, devido ao elevado tempo total de atraso.

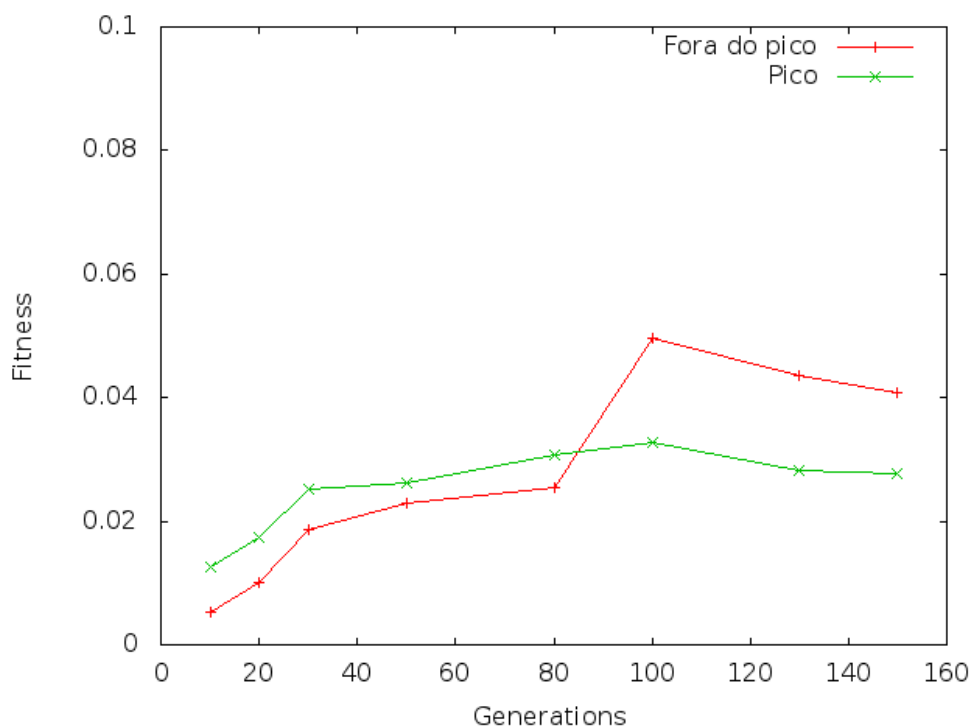


Figura 8.1: Gerações para convergência do Algoritmo Genético.

A Figura 8.2 apresenta o tempo de convergência do algoritmo, é possível observar que o tempo varia quase linearmente com o número de gerações.

## 8.5.3 Resultados: Cenário 1

Considerando um tamanho de população igual a 1.000 e um número de gerações igual a 100, os resultados para essas execuções no Cenário 1 são exibidos na Tabela 8.2. A primeira coluna apresenta a soma do atraso de todos os voos, a segunda coluna apresenta a média de atraso por voo e nas colunas três e quatro são apresentados os parâmetros variados: taxa de mutação e de *crossover*.

A Tabela 8.2 apresenta a análise dos resultados dos voos durante o horário de pico, a maior diminuição no atraso foi obtida com uma taxa de mutação igual a 0,005 e uma taxa de *crossover* igual a 0,7.

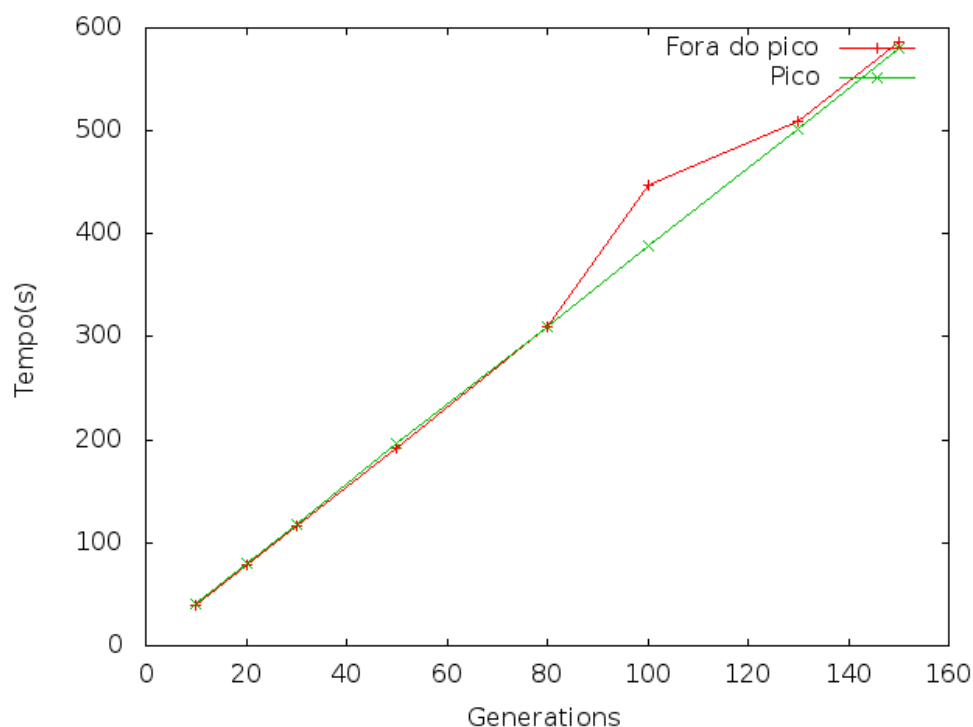


Figura 8.2: Tempo de convergência do Algoritmo Genético.

Tabela 8.2: Resultados obtidos com o modelo proposto no Cenário 1.

Atraso total	Média	Taxa de mutação	Taxa de <i>crossover</i>
17h e 33min	21min	0,01	0,9
14h e 40min	18min	0,005	0,9
17h e 21min	21min	0,01	0,7
14h e 5min	17min	0,005	0,7

#### 8.5.4 Resultados: Cenário 2

Considerando um tamanho de população igual a 1.000 e um número de gerações igual a 100, os resultados para essas execuções no Cenário 2 são exibidos na Tabela 8.3.

Tabela 8.3: Resultados obtidos com o modelo proposto, no Cenário 2.

Atraso total	Média	Taxa de mutação	Taxa de <i>crossover</i>
9h e 40min	13min	0,01	0,9
8h e 2min	11min	0,005	0,9
6h e 32min	9min	0,01	0,7
9h e 43min	13min	0,005	0,7

Na Tabela 8.3 os resultados das análises dos voos fora do horário de pico são apresentados, o melhor resultado para o atraso foi obtido com uma taxa de mutação igual a 0,01 e uma taxa de *crossover* igual a 0,7.

## 8.6 Comparação entre os Dados Reais e os Dados Obtidos.

Cada caso testado com a implementação proposta foi comparado com os casos reais analisados, a Tabela 8.4 exhibe esta comparação. A primeira coluna mostra cada caso, as colunas dois e três mostram os parâmetros variados e a penúltima coluna apresenta a diferença entre o atraso real e o atraso obtido a partir das simulações. A última coluna apresenta a diminuição aproximada do atraso em porcentagem.

Tabela 8.4: Comparação entre os dados reais e os dados obtidos.

Situação	Taxa de mutação	Taxa de <i>crossover</i>	Diferença	Diminuição
Hora de pico	0,01	0,9	3h e 4min	14%
Hora de pico	0,005	0,9	6h e 47min	31%
Hora de pico	0,01	0,7	4h e 16min	19%
Hora de pico	0,005	0,7	7h e 32min	35%
Hora normal	0,01	0,9	1h e 32 min	13%
Hora normal	0,005	0,9	3h e 10min	28%
Hora normal	0,01	0,7	4h e 40min	41%
Hora normal	0,005	0,7	1h e 29min	13%

A Tabela 8.4 mostra que o atraso diminuiu significativamente para alguns casos do horário de pico, a diminuição foi ainda maior para o horário normal. O algoritmo se baseia na ideia de realocar os voos a partir de *slots* cancelados ou atrasados. Fora do horário de pico o número de aeronaves é menor, então existem mais *slots* disponíveis e, assim, existe mais “espaço” para que o algoritmo genético possa realizar modificações, por isso o melhor resultado foi obtido no horário normal.

Os melhores resultados foram obtidos com uma taxa de *crossover* igual a 0,7, isto pode ser explicado porque uma taxa muito alta de *crossover* pode alterar demais a ordem dos *slots*, se afastando das soluções ideais. A taxa de mutação escolhida foi muito baixa e somente dois *slots* poderiam sofrer mutação em cada cromossomo então a mutação não interferiu tanto no resultado.

## 8.7 Resultado da Alocação de *Slots*

É necessário também analisar como foram alocados os *slots*, se houve uma melhoria de *slots* para as empresas que notificaram o atraso. Esta análise foi feita para a alocação que mais diminuiu o atraso no horário de pico, que foi o foco inicial deste trabalho.



A Tabela 8.5 mostra algumas das decolagens do horário de pico, esta é a tabela com os dados reais. A primeira coluna mostra a companhia aérea que realizou o voo, a segunda coluna apresenta o número do voo, a terceira coluna exibe o horário marcado para a decolagem do voo. Na quarta coluna está o horário real de decolagem e na última coluna está o tempo de atraso entre o horário previsto e o horário real. Os voos com atraso superior a 30 minutos foram marcados em vermelhos, estes voos foram considerados fixos.

A Tabela 8.6 mostra os resultados obtidos pelo sistema implementado, a tabela está organizada da mesma forma que a Tabela 8.5. A quarta coluna mostra o horário do *slot* para qual o voo foi alocado. A última coluna apresenta os resultados da diminuição do atraso para cada voo, em poucos casos o atraso do voo aumentou, estes casos são representados por um “-”, os outros casos apresentam a porcentagem de diminuição do atraso.

Pela diferença entre as Tabelas 8.5 e 8.6 é possível perceber que muitos voos da GOL e da TAM foram realocados, pois elas possuíam a maior quantidade de voos atrasados e avisados. No contexto brasileiro é difícil avaliar esta métrica, pois, como a TAM e a GOL realizam a maior parte dos voos, a probabilidade é de que elas terão a maioria dos voos atrasados ou cancelados e também, com maior chance, conseguirão os melhores *slots*.

Esta característica do algoritmo ficaria mais evidente em situações que envolvem um maior número de companhias aéreas em um período de tempo maior.

## 8.8 Comparação com CoDMAN

Conforme foi visto na seção 5.1, o CoDMAN [53] também realiza o sequenciamento de partidas com a alocação de *slots*. É interessante comparar os resultados de ambas as propostas para verificar quais as vantagens e desvantagens de cada uma. Ambos os trabalhos realizam esta alocação de *slots* na fase de pré-voo. O CoDMAN busca diminuir o custo de atraso, que leva em consideração as capacidades de decolagem e aterrissagem dos aeroportos. Considerando apenas a diminuição do atraso obtida pelo CoDMAN, é observado que a maior porcentagem de diminuição foi de 16%. A porcentagem de diminuição do atraso foi maior no presente modelo.

O CoDMAN levou em conta diversos fatores dos aeroportos, como já mencionado anteriormente, as capacidades dos aeroportos, ele considerou restrições não consideradas neste modelo. Pode-se considerar que o modelo da proposta deste trabalho representa um limitante superior para a diminuição do atraso, ou seja, este valor de diminuição ocorre em um cenário ideal, onde todas as empresas concordariam com as modificações realizadas. O CoDMAN é um modelo mais pessimista, ele leva em conta o processo de negociação entre as empresas, ou seja, cada empresa busca a melhor situação para si mesma, em uma situação de competição.

Ambos os modelos são interessantes pois é importante conhecer o máximo possível para uma otimização, ou seja, o caso ideal, mas também é importante se considerar todas as variáveis que podem influenciar no resultando, obtendo uma previsão mais pessimista. Pode ser interessante aplicar o CoDMAN a partir da sequência de *slots* gerada pelo presente trabalho.

Tabela 8.5: Voos no horário de pico.

Companhia Aérea	Voo	Horário Previsto	Horário Confirmado	Atraso
TAM	3739	08:01	08:06	00:05
GOL	1683	08:15	08:20	00:05
AZUL	4427	08:18	08:13	00:00
GOL	1081	08:23	08:44	00:21
GOL	1403	08:29	08:21	00:00
GOL	1144	08:31	08:42	00:11
GOL	1623	08:41	09:01	00:20
GOL	1483	08:50	08:59	00:09
GOL	1405	08:53	09:06	00:13
AVIANCA - ONE	6340	08:55	09:04	00:09
RIO LINHAS AÉREAS	9813	09:00	09:30	00:30
PASSAREDO	2230	09:07	09:23	00:16
AVIANCA - ONE	6189	09:11	09:18	00:07
GOL	1816	09:13	10:21	01:08
GOL	1625	09:20	10:03	00:43
TAM	3821	09:22	09:54	00:32
GOL	1904	09:25	09:50	00:25
GOL	1864	09:36	10:01	00:25
TAM	3025	09:40	10:28	00:48
TAM	3057	09:42	10:17	00:35
AZUL	5019	09:44	09:56	00:12
GOL	1912	09:48	10:14	00:26
GOL	1407	09:50	10:07	00:17
TAM	3540	09:52	10:43	00:51
GOL	1764	09:54	10:52	00:58
GOL	1654	09:56	10:19	00:23
TAM	3840	09:58	10:35	00:37
GOL	1485	10:00	10:32	00:32
TAM	3576	10:08	10:34	00:26
TAM	3860	10:13	10:28	00:15
TAM	3444	10:15	10:27	00:12
GOL	1402	10:17	10:38	00:21
TAM	3482	10:19	10:59	00:40
GOL	1409	10:21	10:53	00:32
TAM	3648	10:23	10:47	00:24
AZUL	4431	10:25	10:33	00:08
TAM	3526	10:27	11:14	00:47
GOL	1083	10:30	10:57	00:27
TAM	3457	10:35	11:18	00:43
TAM	3182	10:37	11:09	00:32
TAM	3703	10:44	11:23	00:39
GOL	1584	10:46	11:15	00:29
GOL	1672	10:48	11:06	00:18
AZUL	4135	10:52	11:02	00:10
TAM	3887	10:54	11:47	00:53
GOL	1685	10:56	11:04	00:08
TAM	3825	10:58	12:25	01:27
TAM	3875	11:00	11:05	00:05
SETE	6414	11:02	11:16	00:14
TRIP	5206	11:06	11:10	00:04

Tabela 8.6: Voos no horário de pico com aplicação da proposta.

<b>Companhia Aérea</b>	<b>Voo</b>	<b>Horário Previsto</b>	<b>Horário Confirmado</b>	<b>Atraso</b>	<b>Diminuição</b>
GOL	1623	8:41	8:38	00:03	85%
AVIANCA - ONE	6340	8:55	8:55	00:00	100%
GOL	1081	8:23	8:18	00:05	76%
AZUL	4427	8:18	8:13	00:05	0%
GOL	1403	8:29	8:18	00:11	-
GOL	1683	8:15	8:13	00:02	60%
GOL	1144	8:31	8:28	00:03	72%
TAM	3739	8:01	8:12	00:11	-
GOL	1483	8:50	8:45	00:05	33%
PASSAREDO	2230	9:07	9:03	00:04	75%
GOL	1405	8:53	8:48	00:05	61%
AVIANCA - ONE	6189	9:11	9:20	00:09	-
RIO LINHAS AÉREAS	9813	9:00	9:30	00:30	0%
GOL	1407	9:50	9:48	00:02	88%
GOL	1864	9:36	9:34	00:02	60%
AZUL	5019	9:44	9:37	00:07	41%
GOL	1904	9:25	9:31	00:06	76%
TAM	3444	10:15	10:13	00:02	83%
GOL	1654	9:56	9:52	00:04	82%
GOL	1912	9:48	9:46	00:02	92%
GOL	1402	10:17	10:13	00:04	80%
GOL	1625	9:20	10:03	00:43	0%
TAM	3860	10:13	10:08	00:05	66%
TAM	3821	9:22	9:54	00:32	0%
TAM	3057	9:42	10:17	00:35	0%
TAM	3025	9:40	10:28	00:48	0%
AZUL	4431	10:25	10:29	00:04	50%
TAM	3576	10:08	10:05	00:03	84%
GOL	1485	10:00	10:32	00:32	0%
AZUL	4135	10:52	10:49	00:03	%
TAM	3540	9:52	10:43	00:51	0%
TAM	3875	11:00	10:58	00:02	60%
GOL	1083	10:30	10:26	00:04	85%
GOL	1672	10:48	10:49	00:01	94%
GOL	1816	9:13	10:21	01:08	0%
GOL	1409	10:21	10:53	00:32	0%
TAM	3182	10:37	11:09	00:32	0%
TRIP	5206	11:06	11:03	00:03	20%
TAM	3648	10:23	10:24	00:01	95%
TAM	3840	9:58	10:35	00:37	0%
GOL	1685	10:56	10:51	00:05	37%
TAM	3526	10:27	11:14	00:47	0%
TAM	3482	10:19	10:59	00:40	0%
TAM	3703	10:44	11:23	00:39	0%
GOL	1764	9:54	10:52	00:58	0%
TAM	3457	10:35	11:18	00:43	0%
SETE	6414	11:02	11:02	00:00	100%
GOL	1584	10:46	10:44	00:02	93%
TAM	3887	10:54	11:47	00:53	0%

# Capítulo 9

## Considerações Finais

Este trabalho é vinculado a um projeto da FINEP com colaboração da equipe do TransLab e apresentou uma proposta para o sequenciamento de partidas de aeronaves com testes realizados a partir de dados do Aeroporto Internacional de Brasília. A proposta deste trabalho foi a utilização de algoritmos genéticos, juntamente com a alocação de *slots* e os conceitos de RBS e *Compression* e sua avaliação é apresentada na Seção 9.1. Por fim, a Seção 9.2 aborda propostas de trabalhos futuros e possibilidades que não puderam ser contempladas por este trabalho.

### 9.1 Avaliação dos Resultados

Atualmente o Aeroporto Internacional de Brasília não utiliza *slots* e nem uma outra técnica para sequenciar a partida de aeronaves, somente o Hotran, que é o horário que a aeronave possui agendado para a partida, porém, como foi visto na análise de resultados, este horário é muitas vezes desrespeitado. A falta de uma gerência de sequenciamento provoca uma desorganização e um baixo aproveitamento da pista. A pista é um recurso escasso, é necessário que ela seja aproveitada ao máximo.

O objetivo da proposta era diminuir o tempo de atraso total dos voos e beneficiar as empresas que informassem os atrasos com antecedência e tais objetivos foram alcançados. Para realizar os testes da proposta, foram utilizados os dados de partidas do Aeroporto Internacional de Brasília, estes dados só fornecem informações básicas dos voos. Para que o trabalho fosse mais completo seria necessário mais informações que não estão disponíveis, como, por exemplo, o motivo de atraso para cada voo, para que fosse analisado se aquele voo está retido ou está somente esperando a pista ser liberada para partir. Para que fosse possível simular os dados em cima da solução proposta foi necessário realizar algumas suposições que podem ter influenciado no resultado final, foi considerado que os voos com atraso maior que 30 minutos não poderiam ser realocados e que as empresas aéreas que possuem estes voos avisaram o fato com antecedência.

Considerando as suposições feitas, os atrasos foram diminuídos significativamente, mas considerando que alguns voos precisam sair atrasados devido a falhas mecânicas ou falhas humanas, é impossível acabar completamente com os atrasos. O fato das empresas aéreas ganharem *slots* melhores para seus outros voos, caso informem um atraso ou cancelamento com antecedência, serve como incentivo para que elas estejam sempre informando a situ-

ação de seus voos para os controladores de tráfego aéreo, para que eles tenham tempo de reorganizar os voos novamente de acordo com a nova alocação.

Em nenhum dos casos testados, o atraso total foi pior que o atraso real, em todos os casos houve uma diminuição, mesmo que mínima, do atraso total.

A eficiência do algoritmo poderia ser melhorada em relação ao tempo com o uso de paralelismo.

Tabela 9.1: Comparação entre os dados reais e os melhores dados obtidos.

Situação	Taxa de mutação	Taxa de <i>crossover</i>	Diferença	Diminuição
Hora de pico	0,005	0,7	7h e 32min	35%
Hora normal	0,01	0,7	4h e 40min	41%

A Tabela 9.1 demonstra o quão eficiente pode ser o algoritmo em duas situações distintas a primeira em horário de pico e a segunda em um horário de funcionamento normal, na primeira houve uma redução de 35% no tempo de atrado e na segunda o algoritmo conseguiu reduzir o tempo de atraso em 41%.

O algoritmo apresentou grande variação nos valores de acordo com as taxas de mutação e *crossover* empregados, mas foi observado uma redução média de aproximadamente 24% tanto em horário de pico como em horário normal.

## 9.2 Trabalhos Futuros

Como trabalho futuros fica a possibilidade de se modificar esta proposta para alocar *slots* de decolagem e aterrissagem de forma simultânea, especialmente para aeroportos que possuem somente uma pista. Também seria interessante sincronizar os *slots* dos aeroportos de partida com os aeroportos de chegada, considerando o tempo entre as rotas de voo.

Outra proposta seria a utilização de outros algoritmos como o Algoritmo Formiga para solucionar esse problema e servir como parâmetro de comparação.

Outros trabalhos do TransLab já implementaram aspectos do controle de solo, estes trabalhos poderiam ser unidos para formar uma única ferramenta de controle de solo.

# Referências

- [1] Departamento de controle do espaço aéreo. <http://www.decea.gov.br>. Acessado em: 13/06/2013.
- [2] Summer Schools Abroad, Rio de Janeiro - RJ. Lecture documentation: Introduction to aircraft performance, 2007.
- [3] O. H. Allen. Fitness and its role in evolutionary genetics. *Nature Reviews Genetics*, 10(8):531–539, 2009.
- [4] ANAC. Anuário do transporte aéreo. Disponível Online. <http://www2.anac.gov.br/arquivos/zip/Anuario2011.zip>, 2011.
- [5] I. Anagnostakis and J.-P Clarke. Runway operations planning: a two-stage solution methodology. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference*, 2003.
- [6] B. André, S. Harmen, and S. Michael. Fitness function design to improve evolutionary structural testing. In *GECCO*, volume 2, pages 1329–1336, 2002.
- [7] H. Balakrishnan. Techniques for reallocating airport resources during adverse weather. In *Decision and Control, 2007 46th IEEE Conference on*, pages 2949–2956, 2007.
- [8] M. Bengi and S. Lance. The impact of ground delay program (GDP) rationing rules on passenger and airline equity. In *International Conference on Research in Air Transportation*, 2008.
- [9] Reuters Brasil. Relatório sobre tráfego aéreo caótico na copa contradiz governo. Disponível Online, Dezembro 2013.
- [10] G. M. Camargo. Controle da pressão seletiva em algoritmo genético aplicado à otimização de demanda em infra-estrutura aeronáutica. Tese de mestrado, Escola Politécnica da Universidade de São Paulo - USP, 2006.
- [11] M. C. Carlos. A importância da regulação de slots aeroportuários para a concorrência no transporte aéreo no Brasil. Master’s thesis, Escola do Tribunal de Contas da União, 2011.
- [12] B. Chris, A. Stephen, C. Lara, and L. Steven. Ration by schedule for airport arrival and departure planning and scheduling. In *Integrated Communications Navigation and Surveillance Conference (ICNS), 2010*, pages I3–1–I3–9, 2010.

- [13] D. W. Christopher. *Flight to the future: human factors in air traffic control*. National Academy Press, 1997.
- [14] R. R. Colin. A genetic algorithm for flowshop sequencing. *Computers & operations research*, 22(1):5–13, 1995.
- [15] R. R. Colin and E. R. Jonathan. *Genetic Algorithms - Principles and Perspectives: A Guide to GA Theory*. Operations Research/Computer Science Interfaces Series. Springer, 2003.
- [16] A. M. F. Crespo. Módulo de avaliação e apoio à decisão: uma aplicação de aprendizagem por reforço no gerenciamento tático do fluxo de tráfego aéreo. Tese de mestrado, Universidade de Brasília, 2010.
- [17] A. M. F. Crespo, C. V. Aquino, B. B. Souza, L. Weigang, A. C. M. A. Melo, and D. P. Alves. Sistema distribuído de apoio à decisão aplicado ao gerenciamento tático do fluxo de tráfego: Caso Cindacta i. *VI Simpósio de Transporte Aéreo*, 2007.
- [18] Comando da Aeronáutica. Diretrizes para implementação de programas de garantia de qualidade nos serviços de tráfego aéreo. Technical report, 2004.
- [19] B. F. David. An overview of evolutionary programming. In *Evolutionary Algorithms*, pages 89–109. Springer, 1999.
- [20] E. G. David and H. H. John. Genetic algorithms and machine learning. *Machine learning*, 3(2):95–99, 1988.
- [21] V. Duong, F. Ferchaud, C. Gavoille, and M. Mosbah. A new slot allocation for ATFM. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, pages 1075–1079, 2004.
- [22] P. Eriksen. Collaborative decision making information management in airports. In *Digital Avionics Systems Conference, 2002. Proceedings. The 21st*, volume 2, pages 7B4–1–7B4–8 vol.2, 2002.
- [23] EUROCONTROL. *Airport CDM Applications - The Guide*, 2003.
- [24] EUROCONTROL. Airport CDM implementation - the manual for collaborative decision making. Technical report, 2010.
- [25] EUROCONTROL. *Airport CDM Implementation - The Manual*, 2012.
- [26] FAA. Federal aviation administration. Disponível Online, Outubro 2013. <http://www.faa.gov/>.
- [27] L. D. George. *Air transportation systems engineering*. Number v. 193 in Air transportation systems engineering. American Institute of Aeronautics and Astronautics, 2001.
- [28] S. Gernot. Grandfather rights in the market for airport slots. *Transportation Research Part B: Methodological*, 44(1):29–37, January 2010.

- [29] Infraero. Empresa brasileira de infraestrutura aeroportuária. Disponível Online, Outubro 2013. <http://www.infraero.gov.br/>.
- [30] Inframérica. Movimentos operacionais aeroporto internacional de brasilia. Disponível Online, Outubro 2013.
- [31] R. Ingo. *Evolutionsstrategien*. Springer, 1978.
- [32] H. H. John. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press, 1975.
- [33] A. C. Arruda Jr. Aprendizagem por reforço aplicada à análise de impacto no controle de fluxo de tráfego aéreo. Master's thesis, Universidade de Brasília - Departamento de Ciência da Computação, 2009.
- [34] A. Kazda and R.E. Caves. *Airport design and operation*. Emerald Group Pub Limited, 2007.
- [35] D. J. Kenneth, M. S. William, and F. G. Diana. *Using genetic algorithms for concept learning*. Springer, 1994.
- [36] S. Kumara, G. David, and K. Graham. Genetic algorithms. In *Search Methodologies*, pages 97–125. Springer, 2005.
- [37] D. Lawrence. *Handbook of genetic algorithms*. 1991.
- [38] P. B. Liu, M. Hansen, and A. Mukherjee. Scenario-based air traffic flow management: Developing and using capacity scenario trees. *Transportation Research Board Annual Meeting*, 2006.
- [39] P. R. Lucas. Utilização de algoritmo genético para gerenciamento de sequenciamento de táxi. Monografia apresentada como requisito parcial para conclusão do curso de licenciatura em computação, Universidade de Brasília - UNB, 2012.
- [40] B. Michael, C. Chien-Yu, H. Robert, and V. Thomas. *Collaborative Decision Making in Air Traffic Management: Current and Future Research Directions*. Springer Berlin Heidelberg, 2001.
- [41] Ministério da Defesa. Comando da Aeronáutica. *Concepção operacional ATM Nacional. DCA 351-2*, 2011.
- [42] Ministério da Defesa. Comando da Aeronáutica. *Plano de implementação ATM Nacional. PCA 351-3*, 2012.
- [43] U. Nico, A. Emile, B. Hans-Jürgen, van L. Peter JM, and P. Erwin. Genetic local search algorithms for the traveling salesman problem. In *Parallel problem solving from nature*, pages 109–116. Springer, 1991.
- [44] Hans-Paul P. S. *Evolution and optimum seeking: the sixth generation*. John Wiley & Sons, Inc., 1993.



- [45] H.H. Hesselink P. van Leeuwen and J.H.T. Rohling. Scheduling aircraft using constraint satisfaction.
- [46] F. Rafael and O. Alessandro. Distribuição de slots em aeroportos: A experiência internacional e a concentração de frequências de vôo no brasil. *Journal of Transport Literature*, 3(2), 2010.
- [47] R. Raúl. *Neutral Networks: A Systematic Introduction*. Springer, 1996.
- [48] A. Robert. The evolution of strategies in the iterated prisoner’s dilemma. *The dynamics of norms*, pages 1–16, 1987.
- [49] L. Shan, C. John-Paul, and B. Cynthia. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science*, 40(1):15–28, 2006.
- [50] A. M. G. Silva. Sistema de simulação acelerado para análise de fluxo de tráfego aéreo. Tese de mestrado, Instituto Nacional de Pesquisas Espaciais - INPE, 2001.
- [51] S. Stroiney, B. Levy, and C. J. Knickerbocker. Departure management: Savings in taxi time, fuel burn, and emissions. In *Integrated Communications Navigation and Surveillance Conference (ICNS), 2010*, pages J2–1–J2–7, 2010.
- [52] Revista AeroEspaço. Editora Imprensa Técnica. Conhecendo o CNS/ATM, 2011.
- [53] F. R. Vitor. Decisão colaborativa com utilização de teoria dos jogos para o sequenciamento de partidas em aeroportos. Dissertação (mestrado em informática), Universidade de Brasília - UNB, 2013.
- [54] T. Vossen and M. O. Ball. Optimization and mediated bartering models for ground delay programs. *Naval Research Logistics*, (53):75–90, 2005.
- [55] T.W.M. Vossen. *Fair Allocation Methods in Air Traffic Management*. University of Maryland, College Park, 2002.
- [56] Darrell Whitley. A genetic algorithm tutorial. 1993.