



ALGORITMO COLÔNIA DE FORMIGAS PARA O PROBLEMA DE POUSO DE AERONAVES EM MÚLTIPLAS PISTAS

Trabalho de Conclusão de Curso

Engenharia da Computação

Matheus Phelipe Alves Pinto
Orientador: Prof. Dra. Eliane Maria Loiola



**UNIVERSIDADE
DE PERNAMBUCO**

**Universidade de Pernambuco
Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação**

MATHEUS PHELIPE ALVES PINTO

**ALGORITMO COLÔNIA DE FORMIGAS
PARA O PROBLEMA DE POUSO DE
AERONAVES EM MÚLTIPLAS PISTAS**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, Outubro de 2022.

Pinto, Matheus Phelipe Alves

Algoritmo Colônia de Formigas para o Problema de Pouso de Aeronaves em Múltiplas Pistas / Matheus Phelipe Alves Pinto. – Recife - PE, 2022.

xv, 69 f. : il. ; 29 cm.

Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) Universidade de Pernambuco, Escola Politécnica de Pernambuco, Recife, 2021.

Orientador: Prof^a. Dr^a. Eliane Maria Loiola.

Inclui referências.

1. Otimização combinatória. 2. Meta-heurística Colônia de Formigas. 3. Problema de Pouso de Aeronaves. I. Título. II. Loiola, Eliane Maria. III. Universidade de Pernambuco.

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 24/10/2022, às 15h00min, reuniu-se para deliberar sobre a defesa da monografia de conclusão de curso do(a) discente **MATHEUS PHELIPE ALVES PINTO**, orientado(a) pelo(a) professor(a) **ELIANE MARIA LOIOLA**, sob título Algoritmo Colônia de Formigas para o Problema do Pouso de Aeronaves em Múltiplas Pistas, a banca composta pelos professores:

CARMELO JOSE ALBANEZ BASTOS FILHO (PRESIDENTE)

ELIANE MARIA LOIOLA (ORIENTADOR)

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada

Aprovada com Restrições*

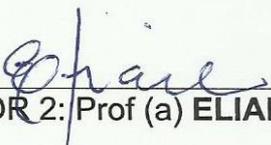
Reprovada

e foi-lhe atribuída nota: 10,00 (Dez)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O(A) discente terá 7 dias para entrega da versão final da monografia a contar da data deste documento.


AVALIADOR 1: Prof (a) **CARMELO JOSE ALBANEZ BASTOS FILHO**


AVALIADOR 2: Prof (a) **ELIANE MARIA LOIOLA**

AVALIADOR 3: Prof (a)

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

*Dedico este trabalho a Deus,
meu Excelso Criador.*

Agradecimentos

Agradeço a Deus pelo dom da vida e pelo seu amparo antes mesmo de nascer. Sem a sua provisão certamente não teria sido capaz de finalizar a minha graduação em Engenharia da Computação. Com o Seu Amor, fui transformado para transformar.

Agradeço aos meus genitores Marluce Alves e Marcello Florentino pelo empenho e dedicação na missão de preparar um cidadão para a vida. Suas lições, testemunhos e sobretudo acompanhamento foram de vital importância para que eu pudesse seguir em frente mesmo quando não havia motivos. O espírito de gratidão se estende a todos os familiares, sem exceção.

Agradeço a minha namorada Kelly Barral pelo seu insondável apoio por meio de palavras de encorajamento, por suas orações nas minhas madrugadas atribuladas. Quando faltaram palavras, seus abraços foram suficientes. Meu coração se quebranta em gratidão a tamanho amor imerecido.

Falar sobre as amizades que construí ao longo dos anos de graduação dispensaria comentários, tornaram meus dias mais leves, ofereceram o devido suporte e me deram motivos para acreditar que o mundo pode ser um lugar melhor. Agradeço a todos que estiveram do meu lado, em especial, Matheus Albert, Michael Matheus, Murilo Stodolni, Nilton Vieira, Richard Jeremias, Pedro Cunha e Selton Guedes. Me sinto feliz e orgulhoso de ter criado esses vínculos, agradeço por se juntarem a mim numa das viagens mais épicas da minha vida.

Por fim, agradeço a todos os professores e funcionários da POLI pelo seu apoio e orientação durante todo o decorrer da minha graduação, em especial à Eliane Maria Loiola, que conduziu a orientação deste trabalho com maestria. Seus ensinamentos certamente serão eternos.

"Mera mudança não é crescimento. Crescimento é a síntese de mudança e continuidade, e onde não há continuidade não há crescimento."

C. S. Lewis

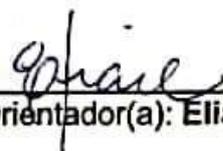
Autorização de publicação de PFC

Eu, **Matheus Phelipe Alves Pinto** autor(a) do projeto de final de curso intitulado: **Algoritmo Colônia de Formigas para o Problema do Pouso de Aeronaves em Múltiplas Pistas**; autorizo a publicação de seu conteúdo na internet nos portais da Escola Politécnica de Pernambuco e Universidade de Pernambuco.

O conteúdo do projeto de final de curso é de responsabilidade do autor.



Matheus Phelipe Alves Pinto



Orientador(a): **Eliane Maria Loiola**

Coorientador(a):



Prof. de TCC: **Daniel Augusto Ribeiro Chaves**

Data: 24/10/2022

Resumo

A meta-heurística colônia de formigas, proposta por Marco Dorigo em 1992, tem sido empregada para a resolução de problemas matemáticos, em especial aos problemas de otimização combinatória que lidam com o escalonamento de recursos finitos como mostram os trabalhos de Bencheikh. O Problema de Pouso de Aeronaves tem como objetivo determinar a melhor ordem de pouso dos aviões dentre as pistas disponíveis. O objetivo desta monografia é descrever a formulação matemática do problema, bem como realizar experimentos seguida da aplicação da análise quantitativa dos resultados obtidos do algoritmo baseado na meta-heurística colônia de formigas na resolução deste problema. Por meio de ciclos de experimentos com dados teóricos, foram encontradas soluções viáveis em um curto espaço de tempo, mantendo a equidade entre as aeronaves escalonadas.

Palavras-chave: Problema de Pouso de Aeronaves, Gerenciamento do Tráfego Aéreo, Meta-heurística, Algoritmo Colônia de Formigas, Otimização Combinatória.

Abstract

The ant colony meta-heuristic, proposed by Marco Dorigo in 1992, has been used to solve mathematical problems, especially combinatorial optimization problems that deal with finite resource scheduling, as shown in Bencheikh's works. The Aircraft Landing Problem aims to determine the best order of landing of planes among the available runways. The objective of this monograph is to describe the mathematical formulation of the problem, as well as to carry out experiments followed by the application of the quantitative analysis of the results obtained from the algorithm based on the ant colony meta-heuristic to solve this problem. Through cycles of experiments with theoretical data, viable solutions were found in a short time, maintaining equity between the scheduled aircraft.

Keywords: Aircraft Landing Problem. Air Traffic Management, Meta-heuristic, Ant Colony Algorithm, Combinatorial Optimization.

Lista de Ilustrações

Figura 1 – Variação do custo para um avião dentro de sua janela de pouso.	20
Figura 2 – Estratégias para agendamento de aeronaves.	21
Figura 3 – Sequência de pouso para o cenário com única pista.	23
Figura 4 – Sequência de pouso para cenário com múltiplas pistas.	24
Figura 5 – Experimento de ponte dupla.	29
Figura 6 – Solução viável do vértice de origem ao vértice de destino.	31
Figura 7 – Fluxograma genérico do ACO.	32
Figura 8 – Uso do feromônio e da heurística na regra de transição.	33
Figura 9 – Pseudocódigo genérico do ACO.	36
Figura 10 – Representação gráfica do problema.	37
Figura 11 – Fluxograma do ACO aplicado ao ALP.	38
Figura 12 – Construção de uma solução pela formiga artificial.	41
Figura 13 – Quantidade de feromônio depositada por <i>slot</i>	48
Figura 14 – Pseudocódigo do ACO aplicado ao ALP.	49
Figura 15 – Evolução dos custos da instância <i>airland_1</i> no Ciclo 1.	56
Figura 16 – Comparativo da evolução dos custos para <i>airland_8</i> no Ciclo 1.	57
Figura 17 – Evolução dos custos da instância <i>airland_2</i> no Ciclo 2.	58
Figura 18 – Evolução dos custos da instância <i>airland_3</i> no Ciclo 2.	58
Figura 19 – Comparativo da evolução dos custos para <i>airland_8</i> no Ciclo 3.	60

Lista de Tabelas

Tabela 1 – Categorias das aeronaves e seus tempos de separação mínimos.	22
Tabela 2 – Notações utilizadas para o Problema de Pouso de Aeronaves.....	26
Tabela 3 – Matriz de adjacência do problema.....	39
Tabela 4 – Matriz de feromônio do ACO.	40
Tabela 5 – Exemplo da estratégia de intensificação.	47
Tabela 6 – Tamanho das instâncias disponíveis.....	51
Tabela 7 – Tempos das aeronaves da instância <i>airland_1</i>	52
Tabela 8 – Tempo de separação entre as aeronaves do exemplo 01.....	53
Tabela 9 – Parâmetros utilizados por ciclo de experimento.	53
Tabela 10 – Comparação dos resultados por ciclo de experimento.....	55
Tabela 11 – Comparativo de resultados entre ACO e GA.....	62

Lista de Símbolos e/ou Siglas

ACA – *Ant Colony Algorithm*

ACO – *Ant Colony Optimization*

ACS – *Ant Colony System*

ALP – *Aircraft Landing Problem*

AS – *Ant System*

ATC – *Air Traffic Control*

DECEA – *Departamento de Controle de Tráfego Aéreo*

IATA – *International Air Transport Association*

FCFS – *First-Comes, First Served*

GA – *Genetic Algorithm*

NP – *Nondeterministic Polynomial*

MIT – *Massachusetts Institute of Technology*

Sumário

1	Introdução.....	14
1.1	Justificativa	15
1.2	Objetivos.....	16
1.2.1	Objetivos específicos.....	16
1.3	Metodologia	16
1.4	Estrutura do trabalho	17
2	O Problema de Pouso de Aeronaves.....	18
2.1	Gerenciamento do Tráfego Aéreo	18
2.2	Categorias de Aeronaves	22
2.3	Cenário com única ou múltiplas pistas de pouso.....	23
3	Modelagem do Problema.....	25
3.1	Formulação do problema.....	25
4	Meta-heurística Otimização por Colônia de Formigas.....	29
4.1	Descrição do algoritmo	30
4.2	Construção da solução	32
4.3	Atualização do feromônio	34
4.4	Informação Heurística.....	35
5	ACO adaptado para Problema de Pouso de Aeronaves.....	37
5.1	Representação do problema	37
5.2	Representação da matriz de adjacência.....	39
5.3	Representação de uma formiga	39
5.4	Representação da matriz de feromônio.....	40
5.5	Representação da matriz heurística	40

5.6	Construção da solução	40
5.7	Seleção da pista de pouso	42
5.8	Informação Heurística.....	43
5.9	Seleção da aeronave.....	44
5.10	Atribuição do tempo de pouso	45
5.11	Atualização da matriz de feromônio.....	46
5.12	Pseudocódigo adaptado para o ALP	49
6	Experimento.....	50
6.1	Descrição das instâncias utilizadas	50
6.2	Descrição do experimento	53
7	Resultados.....	54
7.1	Resultados encontrados	54
7.2	Análise dos resultados.....	55
7.3	Comparativo entre ACO e GA	61
8	Considerações Finais e Trabalhos Futuros.....	63
9	Referências.....	65

1 Introdução

O cenário de transporte aéreo se destaca pela pesquisa e desenvolvimento de soluções tecnológicas robustas e validadas matematicamente tendo em vista o atendimento de exigências que envolvem a adequação, a segurança, a escalabilidade e, sobretudo, o melhor aproveitamento dos recursos financeiros destinados a todos os sistemas em operação (IKLI *et al.*, 2021).

Uma das operações deste setor se refere ao sequenciamento e escalonamento dos tempos de pouso e decolagem de aviões nas pistas de um determinado aeroporto, denotado de Problema do Pouso de Aeronaves (do inglês, *ALP – Aircraft Landing Problem*), o qual envolve tratativas para gerência do intenso fluxo de aterrissagens e decolagens das aeronaves, trazendo uma alta complexidade ao problema em decorrência da existência de uma série de fatores envolvidos, tais como o consumo de combustível, velocidade de voo e aspectos climáticos (IKLI *et al.*, 2021). Em linhas gerais, o maior desafio desta operação reside na gestão eficiente do uso das pistas para decolagem e pouso das aeronaves. Dessa forma, compreende-se que o principal objetivo do problema está na redução dos custos que estão associados aos tempos relativos aos atrasos e adiamentos nos pousos e decolagens das aeronaves (BEASLEY *et al.*, 2000).

Segundo Avella *et al.* (2017), embora o uso de métodos exatos para a resolução do problema para pequenas instâncias seja viável, utilizando a estratégia clássica que faz o uso do algoritmo FCFS (do inglês, *First-Comes, First-Served*), também conhecido como FIFO (do inglês, *First-In, First-Out*) Isto é, a primeira aeronave a chegar no radar do controlador de tráfego aéreo (do inglês, *ATC - Air Traffic Control*) será a primeira a aterrissar, respeitando os tempos limites de separação das aeronaves (um valor mínimo, e um valor máximo permitido). Ainda apontado por Avella *et al.* (2017), além dos altos custos do uso de recursos computacionais para instâncias do mundo real, em muitos cenários tornando sua utilização inviável, também é sabido que tal estratégia não traz resultados ótimos; isto é, não se preocupam em prover soluções com o menor custo e menor tempo total de espera para pouso de todas as aeronaves (BALAKRISHNAN; CHANDRAN, 2006).

Por outro lado, de acordo com Blum e Roli (2003), a utilização de algoritmos computacionais baseados em experiência natural e meta-análises capazes de lidar com problemas de otimização combinatória é cada vez mais notada, proporcionando a possibilidade de se obter uma boa solução, ao invés de uma solução ótima em troca de um tempo de resolução mais eficiente que os algoritmos tradicionais.

Dentre estes algoritmos, a meta-heurística de Otimização por Colônia de Formigas (do inglês, *Ant Colony Optimization - ACO*) tem se mostrado eficiente para resolver problemas de otimização combinatória, em especial para problemas que envolvem o escalonamento de tempo (DORIGO *et al.*, 1991). A ACO consiste em um algoritmo inspirado no comportamento das colônias de formigas, que funciona como um método probabilístico e constrói soluções por meio da inteligência coletiva. Desta maneira, usa a experiência adquirida durante o processo de busca de forma adaptativa.

1.1 Justificativa

De acordo com Beasley *et al.* (2000), O ALP tende a exigir grandes esforços computacionais, em escala exponencial, mesmo para os cenários estáticos. Sendo caracterizado como um problema do tipo NP-Difícil. De acordo com IATA (2017) é estimado que em 2036 cerca de 7.8 bilhões de passageiros realizarão viagens aéreas. A crescente demanda pressiona a infraestrutura, expondo riscos de saturação do espaço aéreo.

Neste trabalho será apresentada a formulação matemática do problema considerando o cenário estático, onde todas as informações são conhecidas, e utilizará a ACO (uma versão adaptada do *Ant Colony System*), o qual foi baseado no trabalho de Bencheikh *et al.* (2011), para a descoberta de uma solução viável que atenda às restrições do problema, aproveitando-se da estrutura de dados baseada no grafo do problema, que fomenta a resolução para o cenário em múltiplas pistas, além de ser capaz de criar soluções estritamente viáveis.

1.2 Objetivos

Realizar um estudo experimental para verificar a viabilidade, eficiência e aplicabilidade do uso da meta-heurística Colônia de Formigas na resolução do Problema do Pouso de Aeronaves.

1.2.1 Objetivos específicos

- Compreender o modelo matemático para solucionar o Problema do Pouso de Aeronave em múltiplas pistas;
- Elaborar um modelo utilizando o algoritmo colônia de formigas aplicado ao problema;
- Analisar a viabilidade do uso do algoritmo a partir dos resultados dos experimentos executados;
- Propor adaptações no algoritmo conhecido para torná-lo mais eficiente e adequado ao cenário em questão.

1.3 Metodologia

Este trabalho tem natureza exploratória e quantitativa (BARROS NETO, 2002). Trata-se de pesquisa para o desenvolvimento de conhecimento na resolução de um problema de otimização combinatória relacionado com tráfego aéreo, portanto explora conceitos da área de negócio, assim como o estudo da modelagem e resolução de problemas de otimização combinatório, e, aplica uma análise quantitativa dos resultados obtidos nos experimentos realizados.

A monografia discutirá os conceitos relacionados ao Problema de Pouso de Aeronaves para o cenário estático, incluindo a formulação matemática adotada. Apresenta a meta-heurística Otimização por Colônia de Formigas, assim como as adaptações necessárias para o uso do ACO na resolução do problema tratado.

1.4 Estrutura do trabalho

Além deste capítulo introdutório, este trabalho se encontra estruturado da seguinte forma: o Capítulo 2 realiza uma apresentação dos principais conceitos que envolvem o problema de escalonamento dos tempos de pouso de aeronaves, incluindo uma breve revisão bibliográfica dos principais tópicos utilizados neste trabalho; o Capítulo 3 apresenta a modelagem do problema; o Capítulo 4 apresenta a meta-heurística Otimização por Colônia de Formigas; enquanto que o Capítulo 5 apresenta a meta-heurística colônia de formigas adaptada ao Problema de Pouso de Aeronaves; o Capítulo 6 apresenta a metodologia utilizada, descreve o conjunto de instâncias utilizadas e as ferramentas utilizadas para construção do algoritmo; o Capítulo 7 apresenta os experimentos realizados e discute os resultados encontrados; e, por fim, o Capítulo 8 apresenta algumas considerações finais e recomendações para trabalhos futuros.

2 O Problema de Pouso de Aeronaves

Neste capítulo são apresentadas as principais terminologias utilizadas ao longo deste trabalho. Serão discutidos os principais conceitos relacionados ao escalonamento de aeronaves em pistas de pouso e a problemática envolvida nos cenários com uma pista e com múltiplas pistas.

2.1 Gerenciamento do Tráfego Aéreo

Sobre o gerenciamento do tráfego aéreo, segundo Leighfisher (2012), em um aeroporto, são definidos os seguintes componentes fundamentais para funcionamento das operações de pouso e decolagem: pista de pouso, pista de táxi, pontos de espera, pátios e boxes de estacionamento. Uma outra terminologia relevante refere-se à alocação de *slots* – que é o horário estabelecido para que a aeronave realize a operação e pouso ou decolagem (DECEA, 2018).

A principal preocupação da gestão do tráfego aéreo é garantir a segurança operacional. Os parâmetros mínimos de separação vertical e horizontal devem sempre ser respeitados. Durante a fase de aproximação, separações de esteira de turbulência são usadas para amortecer os efeitos dos vórtices formados durante o sobrevoo. A esteira de turbulência é um fenômeno potencialmente perigoso que pode fazer com que a aeronave a seguir se torne instável.

Um outro aspecto relevante desta área de negócio é mitigar o congestionamento aeroportuário. O crescimento da demanda por transporte aéreo nas últimas décadas não acompanhou o crescimento da capacidade do sistema aeroportuário no mesmo ritmo. Como resultado, os desequilíbrios entre demanda e capacidade estão se tornando cada vez mais comuns, levando a congestionamentos, atrasos de voos, aumento dos custos das companhias aéreas e insatisfações dos passageiros (GHONIEM *et al.*, 2014).

Este ponto de conexão está localizado principalmente nos principais aeroportos do mundo. Devido à concentração de voos internacionais e domésticos de longa distância, esses locais se tornam importantes pontos de convergência para viagens aéreas e tendem a criar gargalos. Conforme apontado por Trivizas (1998), abordar o ALP é extremamente importante, pois pequenos ganhos na operabilidade da pista, mesmo que inferiores a 5%, podem aumentar a segurança de voo e reduzir o consumo de combustível em períodos de congestionamento aeroportuário.

O Problema do Pouso de Aeronaves tem enorme importância para os controladores de tráfego aéreo em um dado aeroporto. Dado um conjunto de aviões com seus tempos preferíveis de pouso, suas janelas de pouso e os tempos de separação entre os aviões, o Problema do Pouso de Aeronaves tem como objetivo minimizar os desvios dos tempos alvos de pouso de cada aeronave contida no conjunto (BEASLEY *et al.*, 2000). Portanto, trata-se de um problema de otimização em larga escala, ocorrendo em aeroportos movimentados, onde a utilização ótima dos recursos finitos (pistas) é primordial para garantir o bom funcionamento de todo o aeroporto. As aeronaves são separadas seguindo os critérios de equidade (ordem de pouso) e capacidade de cada pista em operação (FURINI *et al.*, 2015).

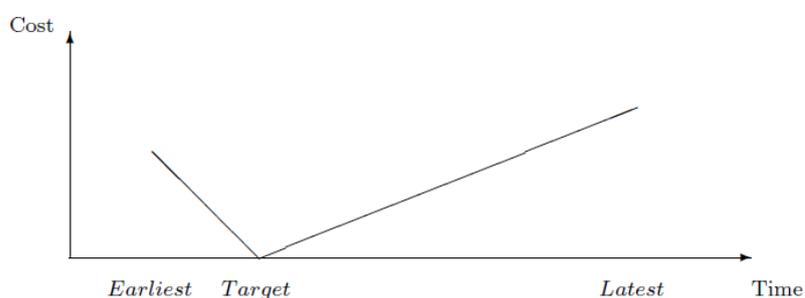
Para reduzir os custos dos desvios, é preciso distribuir os aviões na melhor sequência de pouso possível; atribuindo seus respectivos tempos para um grupo de aeronaves observáveis dentro do alcance do radar, respeitando uma ampla gama de restrições como a janela de pouso permitida e os intervalos de separação entre cada aeronave. Cada aeronave deve pousar em uma das pistas dentro de seu intervalo predeterminado para atender aos critérios de separação entre todos os pares de aeronaves. Dois voos consecutivos de aproximação para pouso devem ser separados por um intervalo de tempo mínimo diretamente relacionado aos tipos de aeronave envolvidos, este tempo é chamado de tempo de separação (AVELLA *et al.*, 2017).

O tempo de pouso deve estar dentro do intervalo de tempo pré-determinado, limitado por um pouso antecipado e um pouso tardio, denominado janela de pouso. Cada aeronave possui intervalos de tempo distintos na escala diária do aeroporto, portanto cada aeronave k tem sua janela de pouso entre t_0^k e t_{max}^k .

O primeiro tempo do intervalo t_0^k representa o tempo de pouso da aeronave que conseguiu executar o voo até o aeroporto em velocidade máxima. E, o tempo máximo de pouso t_{max}^k representa o máximo tempo permitido que uma aeronave pode ficar no ar a espera da disponibilidade para pouso (AVELLA *et al.*, 2017).

Ainda a respeito da janela de pouso, é sabido que cada avião possui uma velocidade de voo mais econômica, conhecida como a velocidade de cruzeiro. Nesse aspecto, o tempo preferido de pouso se refere ao tempo de pouso do avião caso utilize a velocidade de cruzeiro (WEN, 2005). No entanto, considerando o espaço aéreo no qual a aeronave se encontra, o controlador de tráfego aéreo (do inglês, ATC - *Air Traffic Control*) pode exigir que o avião sofra acelerações (antecipando o pouso) ou desacelerações (adiantando o pouso) em consonância com as políticas de segurança (LEIGHFISHER, 2012). A Figura 1 retrata a variação do custo de um avião dentro de sua janela de tempo.

Figura 1 – Variação do custo para um avião dentro de sua janela de pouso.

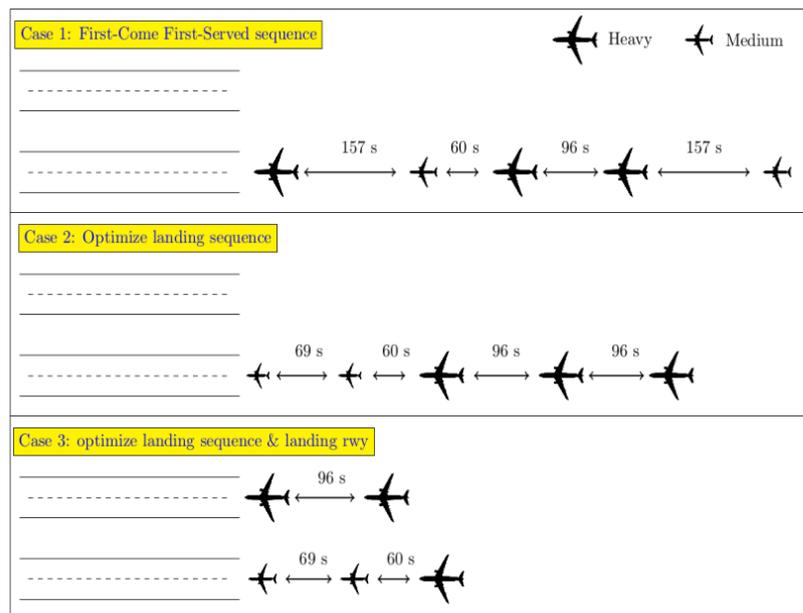


Fonte: Arenales *et al.* (2006).

Para o controlador de tráfego, a maneira mais simples de resolver a problemática faz uso da regra “primeiro a chegar, primeiro a ser servido”. No entanto, tal solução não considera os diferentes tipos de aeronaves, isto é, seus diferentes custos e tempos de separação a serem respeitados, não sendo capaz de promover ganhos na capacidade da pista, principalmente nos cenários com centenas de aeronaves, fornecendo atrasos substanciais e custos elevados para as últimas aeronaves que entram em rota de pouso, como pode ser visto na Figura 2. Portanto, é necessário buscar métodos de resolução que gerem soluções viáveis em um

intervalo de tempo aceitável, respeitando, ainda, os tempos de separação entre cada aeronave, suas respectivas prioridades de pouso e o uso das pistas de pouso disponíveis (IKLI *et al.*, 2021).

Figura 2 – Estratégias para agendamento de aeronaves.



Fonte: IKLI *et al.* (2019).

O ALP pode ser estático ou dinâmico. O problema é dito estático quando todas as informações da aeronave são fixas e não há alteração nas informações quando se inicia o processo de planejamento, o presente trabalho estará lidando com o problema neste contexto. A versão dinâmica do problema, por outro lado, leva em consideração as mudanças dinâmicas nas informações que ocorrem durante o planejamento, quando novas aeronaves podem aparecer dentro do alcance do radar. Na forma dinâmica do problema, variáveis e restrições adicionais de decisão devem ser levadas em consideração, como tempo de chegada de novas aeronaves e tempo de congelamento, indicando que a aeronave está muito próxima da pista e não pode ser escalonada (BEASLEY *et al.*, 2000).

Como ainda apontado por Beasley *et al.* (2000), O problema se caracteriza como um problema de agendamento com restrições, demonstrando similaridades com

outros problemas tais quais como o agendamento de datas de vencimento, de fluxos em lojas, de estações de trabalho e distribuição de lotes.

2.2 Categorias de Aeronaves

Tanto a distância quanto o tempo de separação entres as aeronaves variam de acordo com a categoria das aeronaves. Esta diferença ocorre por questões técnicas como será explicado a seguir.

Segundo o Departamento de Controle do Espaço Aéreo – DECEA (2018) os aviões, quando em movimento, fazem com que o ar atinja suas asas e desta forma os vórtices circulares criados pelo ar em movimento aparecem devido à diferença de pressão nas asas, onde a pressão mais baixa na superfície superior e a pressão mais alta na superfície superior e superfícies inferiores, a coleção de vórtices é chamada de perturbação da esteira. As esteiras de turbulência podem causar capotamentos violentos, perda de altitude ou velocidade de subida. Eles devem ser considerados um grande perigo se houver um encontro de vórtices na área onde uma aeronave está se aproximando de outra, pois a aeronave atrás pode experimentar instabilidade na velocidade, empuxo, altitude e tempo de reação (DECEA, 2020).

Tabela 1 – Categorias das aeronaves e seus tempos de separação mínimos.

		Aeronave seguidora		
		Pesada	Média	Pequena
Aeronave líder	Pesada	94	114	167
	Média	74	74	138
	Pequena	74	74	98

Fonte: Beasley *et al.* (2000).

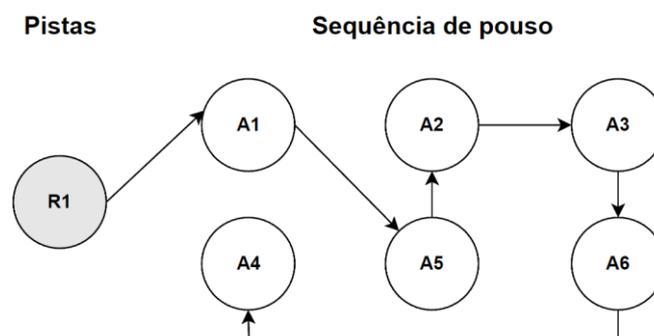
Assim, as aeronaves são divididas em categorias: pesadas, médias e pequenas. Tornando possível determinar de forma segura a separação mínima entre as aeronaves, em distância ou em tempo. A Tabela 1 exemplifica as distâncias mínimas, em segundos, para diferentes pares de aeronaves operando na mesma pista. Segundo o DECAE (2020), os pesos das aeronaves são categorizados da seguinte forma:

- PESADA (H): aeronaves de peso máximo de decolagem de 136.000 kg (300.000 libras) ou mais;
- MEDIA (M): aeronaves de peso máximo de decolagem inferior a 136.000 kg (300.000 libras) e superior a 7000 kg (15.500 libras);
- LEVE (L): aeronaves de peso máximo de decolagem de 7000 kg (15.500 libras) ou menos.

2.3 Cenário com única ou múltiplas pistas de pouso

De acordo com o DECEA (2015), o conceito de saturação aeroportuária é definido como a situação em que a demanda de tráfego aéreo excede a capacidade teórica da pista ou a capacidade de um setor específico de gestão do espaço aéreo. As condições de saturação podem ser compreendidas de forma ilustrativa, em especial no que tange ao custo associado decorrente do escalonamento das aeronaves. Considerando um cenário de agendamento de 6 aeronaves e 2 pistas de pouso, a condição mencionada se mostra latente mediante comparação dos cenários com única e múltiplas pistas. A Figura 3 descreve a sequência de pousos viável considerando a disponibilidade de uma única pista. A partir de uma pista, é criada a sequência de pouso conectado as aeronaves, adicionando os tempos de separação requisitados.

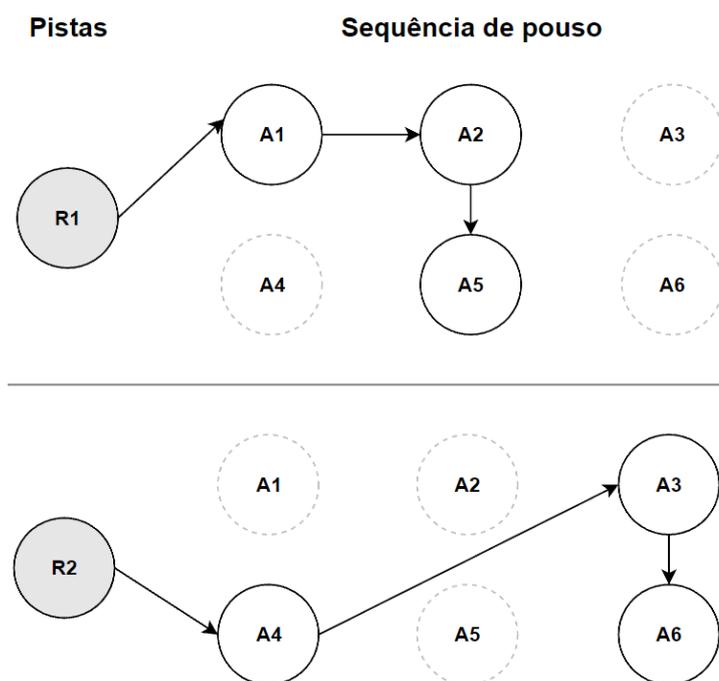
Figura 3 – Sequência de pouso para o cenário com única pista.



Fonte: O autor.

De forma análoga à análise da Figura 3, a Figura 4 demonstra como as aeronaves podem ser alocadas em diferentes pistas de forma a reduzir os custos dos atrasos.

Figura 4 – Sequência de pouso para cenário com múltiplas pistas.



Fonte: O autor.

Além dos tempos de separação entre o pouso de aeronaves, um outro fator que corroboram para a saturação das pistas de pouso refere-se ao tempo de ocupação das pistas, em que a pista está em uso durante a decolagem, calculado a partir do momento em que a aeronave sai do porão até o momento em que cruza a cabeceira oposta. Embora o uso de mais pistas possa prover uma melhor distribuição dos tempos de utilização, bem como reduzir os custos dos pousos das aeronaves, é sabido que se trata de um recurso finito (DECEA, 2015).

3 Modelagem do Problema

Neste capítulo será apresentada a formulação do Problema do Pouso de Aeronaves originalmente proposta por Beasley *et al.* (2000).

Soykan (2016) classifica o ALP na classe de problemas NP-difícil, pois, além da complexidade computacional inerente ao problema, o tamanho do problema leva em consideração a incerteza e diversos objetivos conflitantes. Portanto, não há algoritmo conhecido para encontrar a solução ótima para um problema de tamanho real em tempo polinomial.

Por se tratar de um problema inerentemente combinatório, pode-se concluir que o uso de métodos exatos para a sua resolução são impraticáveis, em virtude das dimensões do problema. Sendo assim, se faz necessária a busca de outros métodos que sejam capazes de prover soluções viáveis em tempos computacionais razoáveis. A partir das heurísticas e meta-heurísticas, tais resultados podem ser alcançados, mesmo tendo ciência de que não será possível garantir que os resultados encontrados pelas heurísticas sejam ditos ótimos (BLUM; ROLI, 2003).

3.1 Formulação do problema

A formulação do Problema do Pouso de Aeronaves utiliza variáveis binárias e variáveis contínuas, portanto, trata-se de uma formulação mista.

A Tabela 2 destaca as principais notações utilizadas para se referir às informações do Problema do Pouso de Aeronaves.

Na sequência, são apresentadas as variáveis de decisão do problema, seguida pela formulação do problema que será utilizada neste trabalho. Portanto, são apresentados a função objetivo e suas restrições, com uma breve descrição que de cada uma delas, para tornar mais fácil a compreensão dos conceitos envolvidos na formulação apresentada.

Tabela 2 – Notações utilizadas para o Problema de Pouso de Aeronaves.

m	Quantidade de pistas de pouso ($m \in N^*$).
n	Quantidade de aeronaves ($n \in N^*$).
K	Conjunto de pistas para pouso ($K = \{1, 2, \dots, m\}$).
A	Conjunto de aeronaves esperando pouso ($A = \{1, 2, \dots, n\}$).
E_i	Tempo mínimo de pouso da aeronave ($i \in A$).
L_i	Tempo máximo de pouso da aeronave ($i \in A, L_i > E_i$)
Ta_i	Tempo alvo de pouso da aeronave ($i \in A$).
x_i	Tempo de pouso atribuído ($i \in A, E_i \leq x_i \leq L_i$).
x_i^-	Tempo total de adiantamento
x_i^+	Tempo total de atraso
C_i^-	Custo de penalidade unitário para a aeronave i , se a aeronave i pousar antes do tempo alvo.
C_i^+	Custo de penalidade unitário para a aeronave i , se a aeronave i pousar após do tempo alvo.
S_{ij}	Tempo de separação entre a aeronave i e j . Se a aeronave i pousar antes da aeronave j na mesma pista ($S_{ij} > 0, i \neq j$).
s_{ij}	Tempo de separação entre a aeronave i e j . Se a aeronave i pousar antes da aeronave j em pistas diferentes ($s_{ij} > 0, i \neq j$).

Fonte: Adaptado de Bencheikh *et al.* (2011).

Os custos de penalidade C_i^- e C_i^+ representam os custos unitários para a aeronave i que pousa, respectivamente, com antecedência e atraso em relação ao tempo alvo (ambos devem ser ≥ 0).

As equações (1), (2) e (3) descrevem as variáveis binárias do problema.

$$a_{ik} = \begin{cases} 1, & \text{se a aeronave } i \text{ for pousar na pista } k; \\ 0, & \text{caso contrário.} \end{cases} \quad (1)$$

$$Z_{ij} = \begin{cases} 1, & \text{se a aeronave } i \text{ e } j \text{ pousam na mesma pista } (i, j \in A: i \neq j); \\ 0, & \text{caso contrário.} \end{cases} \quad (2)$$

$$\delta_{ij} = \begin{cases} 1, & \text{se a aeronave } i \text{ pousa antes de } j \text{ (} i, j \in A: i \neq j \text{);} \\ 0, & \text{caso contrário.} \end{cases} \quad (3)$$

A redução dos custos de pouso da sequência de aeronaves deve respeitar as duas principais restrições do problema, dadas por (4) para definir que o tempo de pouso de cada avião está contido em sua respectiva janela de tempo, e, por (5) que garante o respeito aos intervalos de separação da aeronave j , que aterrissa após a aeronave i na mesma pista.

$$x_i \in [E_i, L_i] \quad \forall i \in A \quad (4)$$

$$x_j \geq x_i + S_{ij} \in [E_i, L_i] \quad \forall i, j \in A, i \neq j \quad (5)$$

A função objetivo é apresentada em (6) e busca minimizar a soma dos custos de desvios do tempo alvo (Ta_i), respeitado um conjunto de restrições apresentado a seguir:

$$\min Z = \sum_{i \in A} C_i^- x_i^- + C_i^+ x_i^+ \quad (6)$$

Sujeito a:

$$E_i \leq x_i \leq L_i \quad \forall i \in A \quad (7)$$

$$0 \leq x_i^- \leq Ta_i - E_i \quad \forall i \in A \quad (8)$$

$$0 \leq x_i^+ \leq L_i - Ta_i \quad \forall i \in A \quad (9)$$

$$x_i^- \geq Ta_i - x_i \quad \forall i \in A \quad (10)$$

$$x_i^+ \geq x_i - Ta_i \quad \forall i \in A \quad (11)$$

$$x_i = Ta_i - x_i^- + x_i^+ \quad \forall i \in A \quad (12)$$

$$\delta_{ij} + \delta_{ji} = 1 \quad \forall i, j \in A, i \neq j \quad (13)$$

$$\sum_{k \in K} a_{ik} = 1 \quad \forall i \in A \quad (14)$$

$$Z_{ij} = Z_{ji} \quad \forall i, j \in A, i < j \quad (15)$$

$$Z_{ij} \geq a_{ik} + a_{jk} - 1 \quad \forall i, j \in A, i < j, k \in K \quad (16)$$

$$x_j \geq x_i + S_{ij} * z_{ij} + s_{ij}(1 - z_{ij}) - M\delta_{ji} \quad \forall i, j \in A, i < j \quad (17)$$

$$a_{ik}, z_{ij}, \delta_{ji} \in \{0,1\} \quad \forall i, j \in A, i \neq j, k \in K \quad (18)$$

A restrição descrita em (7) garante que o tempo de pouso atribuído para o avião i está dentro da janela de tempo de pouso permitida. As restrições descritas em (8) e (9) garantem que os tempos adiantamento e atraso estejam contidos no intervalo definido, respectivamente. As restrições descritas em (10) e (11) definem os tempos mínimos de adiantamento e atraso. A restrição definida por (12) define o tempo de pouso para a aeronave i . A restrição descrita em (13) garante que a precedência de pouso entre as aeronaves i e j seja respeitada. A restrição descrita em (14) garante que um avião pouse somente em uma pista. A restrição descrita em (15) garante a simetria a da ordem de pouso das aeronaves i e j . A restrição dada por (16) garante que a variável Z_{ij} seja igual a 1 para os casos em que os aviões pousem na mesma pista. A restrição descrita em (17) garante que para duas aeronaves i e j que pousam na mesma pista ou em pistas diferentes (característica mutuamente exclusiva), pelo menos S_{ij} , s_{ij} (respectivamente) unidades de tempo devem ser transcorridas pela aeronave j antes de pousar, garantindo que o tempo de separação entre as aeronaves será respeitado. Por fim, a restrição dada por (18) define as restrições de domínio das variáveis de decisão.

$$M = Li + S_{ij} + Ti \quad \forall i, j \in A, i \neq j \quad (19)$$

$$x_j \geq x_i + S_{ij} * z_{ij} + s_{ij}(1 - z_{ij}) \quad (20)$$

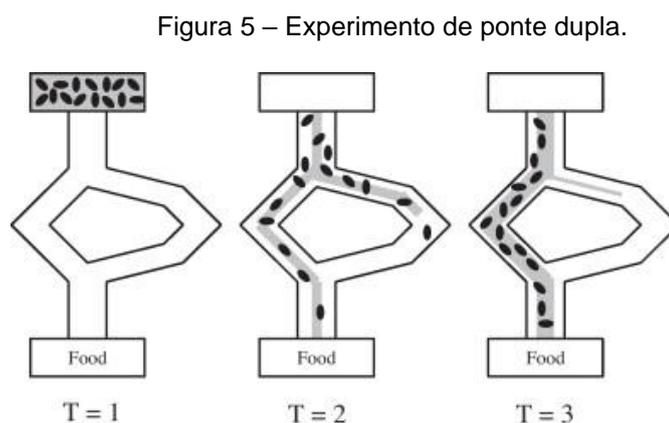
$$x_j \geq x_i + S_{ij} * z_{ij} + s_{ij}(1 - z_{ij}) - M \quad (21)$$

A matriz M , contida em (17) é calculada de acordo com (19). Tal valor é necessário para o caso em que a aeronave j se antecipe e decole antes da aeronave i . O tempo de decolagem agendado x_j será $< E_i$, representando o tempo mínimo inicial de sua decolagem. Se $\delta_{ij} = 1$, então $\delta_{ji} = 0$. Ou seja, o avião i pousará antes de j . O tempo x_j é calculado conforme ilustra a equação (20). Por outro lado, se $\delta_{ij} = 0$, então $\delta_{ji} = 1$. Ou seja, o avião j pousará antes de i . O tempo x_j é calculado de acordo com (21).

4 Meta-heurística Otimização por Colônia de Formigas

Neste capítulo, os fundamentos da meta-heurística Otimização por Colônia de Formigas são apresentados. O campo dos algoritmos de formigas estuda modelos derivados da observação do comportamento de formigas e usa esses modelos como fonte de inspiração para desenvolver novos algoritmos para resolver problemas de otimização e controle distribuído (CUNHA *et al.*, 2012).

O complexo comportamento social de colônias de insetos, como formigas, abelhas e cupins, intrigou as pessoas o suficiente para estudá-las. As formigas no mundo real utilizam um tipo específico de comunicação indireta entre indivíduos conhecido como comunicação de feromônio (BENCHEIKH *et al.*, 2011). A partir do experimento da Ponte Dupla descrito por Goss *et al.* (1989), percebeu-se que o comportamento coletivo da colônia foi capaz realizar o menor percurso entre o ninho e a fonte de alimento. O experimento dispunha de duas rotas de tamanhos diferentes para as formigas transitarem. Ao longo dos ciclos de iterações, a ponte de menor tamanho abrigou a maior concentração de formigas, como pode ser visto na Figura 5.



Fonte: Goss *et al.* (1989).

A partir desses estudos, surgiram os primeiros algoritmos que modelavam o comportamento alimentar complexo que fazem uso da troca de informações de

indivíduos menos complexos que compõem o sistema. A Otimização por Colônia de Formigas (do inglês, ACO - *Ant Colony Optimization*) foi apresentada inicialmente por Dorigo (1992), inspirado pelo comportamento de coleta de alimentos das formigas na natureza, visto na Figura 5. Sendo assim, o ACO possui uma quantidade de formigas artificiais, que compõe uma colônia, assim como no mundo real e que se encarregam de encontrar a solução ótima para um dado problema combinatório, sendo capazes de trocar informações através de feromônios (DORIGO; STÜTZLE, 2004).

A ideia básica é que os princípios de auto-organização que permitem a coordenação do comportamento real das formigas possam ser usados para coordenar populações de agentes cooperantes para resolver problemas computacionais. Vários aspectos diferentes do comportamento das colônias de formigas inspiraram vários algoritmos de colônias de formigas. Por exemplo, divisão de trabalho, seleção de filhotes e transporte público. Em cada um desses exemplos, as formigas coordenam suas atividades usando a estigmergia, uma forma de comunicação mediada por mudanças no ambiente (DORIGO; STÜTZLE, 2004, p. 1).

Em 1991, o *Ant System (AS)* foi criado, sendo o algoritmo pioneiro a fazer uso da meta-heurística, sendo aplicado para resolução do Problema do Caixeiro Viajante com o objetivo de encontrar o menor caminho Hamiltoniano do grafo completo (DORIGO *et al.*, 1991; DORIGO, 1992). Para esse problema, cada formiga na colônia pode construir o seu próprio caminho; a inclusão do feromônio ocorre de acordo com a qualidade da solução encontrada pela formiga. Ou seja, o menor caminho no grafo terá a maior quantidade de feromônio.

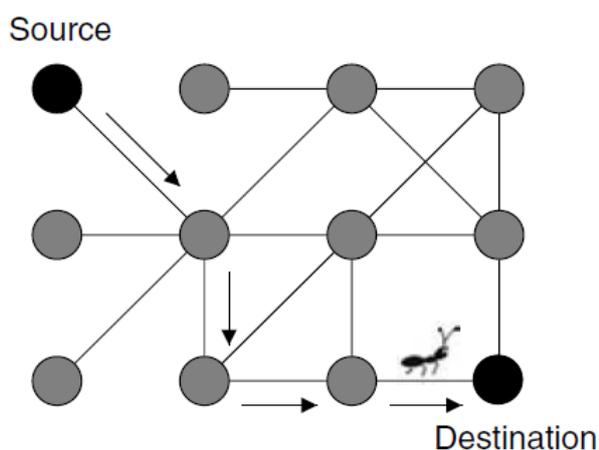
4.1 Descrição do algoritmo

Durante a construção de uma solução, as formigas podem caminhar de duas maneiras: para frente e para trás. Elas estão no modo de avanço ao passar do ninho para a comida e no modo reverso quando estão fazendo o percurso de volta para o ninho. Quando uma formiga chega ao seu destino (ninho ou fonte de alimento), ela muda seu modo de operação. Assim, as formigas que avançam em direção a fonte de alimento constroem uma solução por meio de escolhas probabilísticas para que o

próximo nó seja selecionado dentre aqueles na vizinhança do nó do grafo em que estão localizados.

Conforme ilustrado na Figura 6, a escolha da probabilidade é influenciada por traços de feromônios anteriores que foram depositados no grafo por outras formigas.

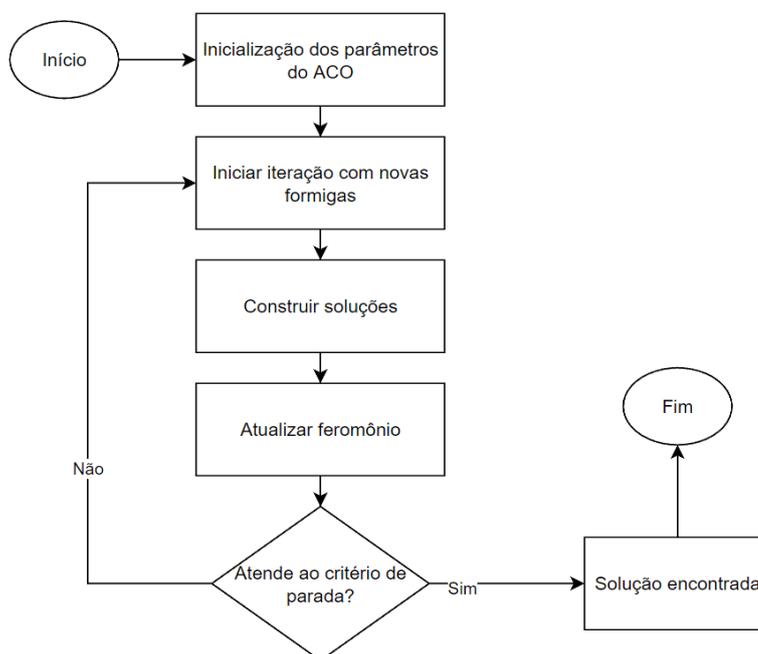
Figura 6 – Solução viável do vértice de origem ao vértice de destino.



Fonte: Dorigo e Stützle (2004).

Dentre as variações das técnicas de diversificação, o *Ant Colony System (ACS)* faz uso da premissa de que as formigas que caminham para frente não depositam feromônios enquanto se movem, enquanto as formigas que retornam para o ninho, depositam feromônio em cada aresta do grafo de forma proporcional a qualidade da escolha daquele caminho. Embora existam variações, o fluxograma genérico da meta-heurística se encontra descrito na Figura 7.

Figura 7 – Fluxograma genérico do ACO.



Fonte: Baseado em Dorigo e Stützle (2004).

4.2 Construção da solução

Dado um grafo $G(V, E)$ cujas arestas i, j representam o problema, o algoritmo conecta cada aresta do grafo ao caminho do feromônio artificial e coloca cada formiga artificial em um vértice específico (nó).

Durante o processamento, as formigas escolhem o próximo nó aleatoriamente seguindo o caminho de acordo com a fórmula de probabilidade em função do feromônio, de modo que as informações de borda são usadas para tomar decisões e a quantidade inicial de feromônio é constante em todos os arcos ou bordas do diagrama (DORIGO; STÜTZLE, 2004).

A equação (22) define a regra de transição probabilística para escolha do próximo nó a ser visitado durante a fase construtiva do algoritmo. Assim, é definida a probabilidade da formiga k percorrer o caminho (i, j) .

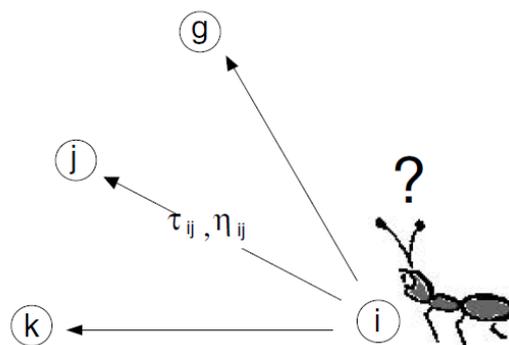
$$p_{ij}^k = \begin{cases} \frac{[\tau_{ij}]^\alpha [n_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [n_{il}]^\beta} & \text{Se } j \in N_i^k \\ 0 & \text{Se } j \notin N_i^k \end{cases} \quad (22)$$

Onde

- τ_{ij} define a quantidade de feromônio na trilha;
- n_{ij} define um valor heurístico conforme as características do problema;
- α é um parâmetro que determina a influência da trilha de feromônio;
- β é um parâmetro que controla a informação heurística;
- l define o conjunto de possíveis soluções;
- N_i^k é o conjunto de nós de vizinhança viável do nó i , associado k – ésima à formiga, ou seja, o conjunto de nós que a formiga ainda não visitou. A probabilidade de uma formiga escolher um nó que não pertence a N_i^k é zero.

Sendo assim, a regra de transição deve ser aplicada durante a fase construtiva, na qual cada formiga utilizará os valores de feromônio e da heurística para decidir a probabilidade de adição de uma determinada aresta em relação à sua vizinhança, como pode ser visto na Figura 8.

Figura 8 – Uso do feromônio e da heurística na regra de transição.



Fonte: Dorigo e Stützle (2004).

4.3 Atualização do feromônio

Conforme apontam Dorigo e Stützle (2004), a atualização do feromônio tem como intuito prover a intensificação das trilhas percorridas de acordo com a qualidade da solução gerada, bem como fornecer um mecanismo de diversificação para evitar que o algoritmo fique limitado a ótimos locais. A seguir, serão discutidas as fases de intensificação e diversificação que a dinâmica do feromônio traz à meta-heurística.

Intensificação: depósito do feromônio

A colônia também possui uma memória coletiva que serve de apoio para a tomada das decisões das formigas ao longo das iterações. Os valores de feromônio armazenados nesta memória podem ser interpretados como uma espécie de tendência da colônia de formigas para escolha de um dado caminho, com base no conhecimento obtido por cada formiga nas iterações passadas.

A atualização dessa memória é feita quando as formigas voltam para o ninho, depositando doses de feromônios em arcos usados durante a construção da solução somente nos nós de maior qualidade, isto é, que melhor se adequam a função objetivo associada. Sendo assim, tornar a função de atualização de feromônios dependente da qualidade da solução produzida pode ajudar futuras formigas a chegar a melhores soluções.

A trilha de feromônio somente será bem aproveitada se cada formiga fizer o uso de uma memória individual (lista tabu) durante a construção da sua solução. Tal artifício explicitamente impede que uma formiga refaça o mesmo caminho que percorreu enquanto procurava o nó de destino. A partir dessa memória, cada formiga é capaz de armazenar não só o caminho percorrido, mas também o custo associado a cada aresta selecionada. Tais informações são fundamentais para a geração de soluções viáveis além de auxiliar no processo de intensificação das trilhas percorridas, que é visível por toda a colônia.

Diversificação: evaporação do feromônio

Em colônias verdadeiras, a intensidade do feromônio diminui com o tempo. No ACO, a evaporação é simulada aplicando a regra de evaporação escolhida corretamente, por exemplo, podendo diminuir gradativamente de acordo com uma

taxa pré-definida. A evaporação reduz o efeito do depósito de feromônios nos estágios iniciais do estudo, onde as formigas artificiais podem ter produzido soluções de baixa qualidade. Dessa forma, esse mecanismo ajuda a eliminar os ótimos locais, aumentando a diversificação das soluções.

A equação (23) ilustra como os mecanismos de intensificação e diversificação são aplicados ao ACO.

$$\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad \forall i \in A, \rho \in [0,1] \quad (23)$$

Onde

- ρ é o coeficiente de evaporação ($\rho < 1$). É utilizado para evitar a convergência de soluções em um espaço de busca local;
- $\tau_{ij}(t + 1)$ é o valor do feromônio atualizado na aresta (i, j) ao término da iteração;
- $\Delta\tau_{ij}(t)$ é a quantidade de feromônio depositada na aresta (i, j) pela colônia ao término da iteração.

4.4 Informação Heurística

Durante a fase de construção de uma solução, as formigas fazem uso de probabilidades para definirem os próximos nós a serem percorridos. Tal escolha é feita baseando-se no conhecimento coletivo da colônia (rastros de feromônio) mas também de informações que indicam o quão conveniente é tomar esse caminho como parte da solução do problema. Tais informações são comumente chamadas de informação heurística (n) (DORIGO; STÜTZLE, 2004).

No ACO, a informação heurística pode assumir duas condições: cenário estático ou dinâmico. Para o primeiro, as informações são computadas ainda na fase de inicialização do algoritmo, permanecendo fixas ao longo das iterações (estratégia útil para determinados problemas, por exemplo, Problema do Caixeiro Viajante, onde as distâncias entre as cidades são imutáveis). Quanto ao segundo, a informação heurística depende dos resultados parciais da construção da solução, sendo

necessária à sua computação após cada passo das formigas, útil para os problemas de escalonamento de tempo, que requerem o recálculo dos tempos das variáveis de decisão associadas. Tal abordagem se demonstra custosa computacionalmente se comparada ao cenário estático. Felizmente, o custo associado à computação é compensado visto que tal abordagem fornece uma maior precisão para a tomada de decisão das formigas.

A Figura 9 mostra o pseudocódigo da meta-heurística Otimização por Colônia de Formigas, o qual pode ser aplicada para solucionar problemas de otimização.

Figura 9 – Pseudocódigo genérico do ACO.

algoritmo ACO

1. Inicializar os parâmetros (τ , n , colônia);
2. Enquanto não alcançar a condição de parada, faça:
 - 2.1. Formigas constroem soluções;
 - 2.2. Atualizar o feromônio;
3. Fim do enquanto;

Fonte: Baseado em Dorigo e Stützle (2004).

Portanto, conclui-se que o uso da informação heurística é de fundamental importância para a construção da heurística aplicada ao problema. A sua natureza estática ou dinâmica está intimamente associada à característica do problema a ser solucionado. É sabido que tal informação pode ser associada a um mecanismo de busca local, melhorando significativamente a performance do algoritmo.

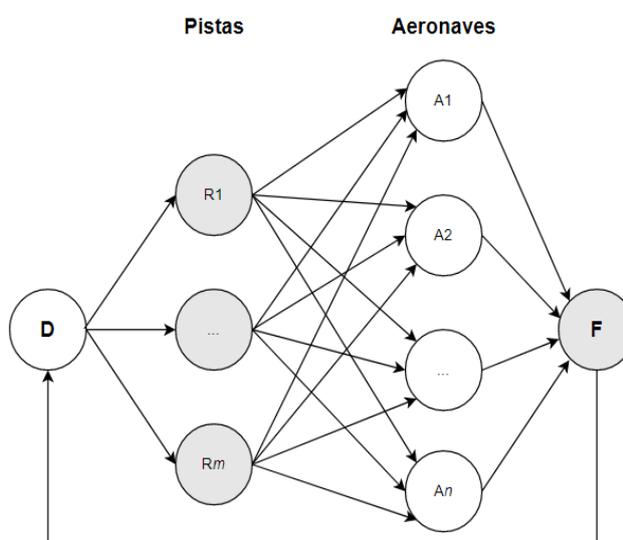
5 ACO adaptado para Problema de Pouso de Aeronaves

Neste capítulo serão apresentadas as adaptações necessárias do ACO para resolução do Problema de Pouso de Aeronaves, o qual é baseada na implementação inicial proposta por Bencheikh *et al.* (2011).

5.1 Representação do problema

Para resolver o Problema de Pouso de Aviões usando ACO será apresentado neste trabalho uma representação baseada em um grafo bipartido, também proposta por Bencheikh *et al.* (2011). No primeiro nível são definidas as pistas disponíveis e no segundo nível os aviões candidatos para pouso. Foram adicionados dois nós fictícios D e F à entrada e à saída do gráfico, respectivamente, cuja ilustração se encontra na Figura 10. Para construir uma solução, uma formiga inicia sua trajetória a partir do nó D , e, primeiramente escolhe uma pista entre as disponíveis.

Figura 10 – Representação gráfica do problema.

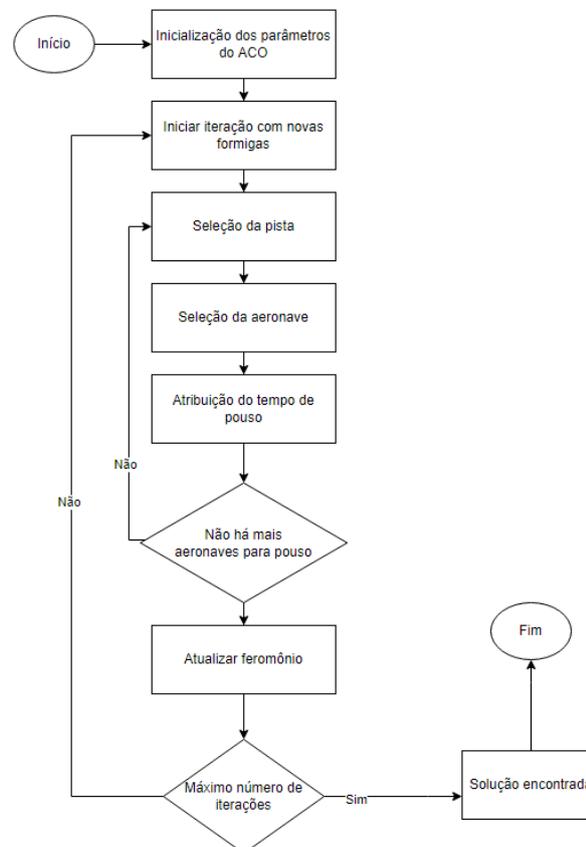


Fonte: Adaptado de Bencheikh *et al.* (2011).

Conforme descrito no fluxograma da Figura 11, uma vez selecionada a pista, segue-se a seleção de uma dessas aeronaves que foram avistadas na verificação do

radar e o momento de seu pouso. Finalmente, a formiga chega ao final do grafo. Antes de selecionar a próxima aeronave, a formiga recalcula os tempos de pouso das aeronaves candidatas, congelando-os e aplicando uma busca local que visa obter a aeronave que produzirá os menores custos de penalidade. Este processo é repetido até que todas as aeronaves tenham pousado. Cada novo ciclo de iteração é composto por novas formigas, com suas respectivas matrizes e aeronaves candidatas resetadas.

Figura 11 – Fluxograma do ACO aplicado ao ALP.



Fonte: Adaptado de Bencheikh *et al.* (2011).

5.2 Representação da matriz de adjacência

A partir da representação do problema através do grafo da Figura 10, é extraída a sua respectiva matriz de adjacência. A partir dela, serão definidas as matrizes de feromônio e da informação heurística da colônia.

Tabela 3 – Matriz de adjacência do problema.

	D	R_1	...	R_m	A_1	...	A_n	F
D	0	1	1	1	0	0	0	0
R_1	0	0	0	0	1	1	1	0
...	0	0	0	0	1	1	1	0
R_m	0	0	0	0	1	1	1	0
A_1	0	0	0	0	0	0	0	1
...	0	0	0	0	0	0	0	1
A_n	0	0	0	0	0	0	0	1
F	1	0	0	0	0	0	0	0

Fonte: O autor.

5.3 Representação de uma formiga

Para resolução do problema, também é necessário que seja adotada uma representação para as formigas. Em especial, para resolução do ALP uma formiga deve possuir uma lista de aeronaves candidatas para pouso; tal lista será utilizada para criar um espécime de “memória” da formiga, impedindo que ela selecione o mesmo avião para pistas distintas, o que levaria a uma solução inviável. Além disso, é necessário possuir uma lista que armazena os resultados para cada pista; cada pista conterà uma possível sequência de pouso das aeronaves, nesse caso, são armazenados os índices dos aviões, e o custo de penalidade total que é calculada por meio da função objetivo (6) para composição da solução (BENCHEIKH *et al.*, 2016).

5.4 Representação da matriz de feromônio

Considerando os conjuntos de pistas e aeronaves, então é formulada a seguinte matriz de feromônios do algoritmo. O par (R_i, A_j) representa a quantidade de feromônio adicionada para o agendamento da aeronave A_j na pista R_i . Quanto maior o valor do feromônio, maior será a probabilidade desta aeronave ser priorizada na fila de agendamento pelas formigas da colônia.

Tabela 4 – Matriz de feromônio do ACO.

	A_1	A_2	...	A_n
R_1	0.5	0.5	...	0.9
R_2	0.2	0.1	...	0.7
...
R_m	0.3	0.6	...	0.4

Fonte: O autor.

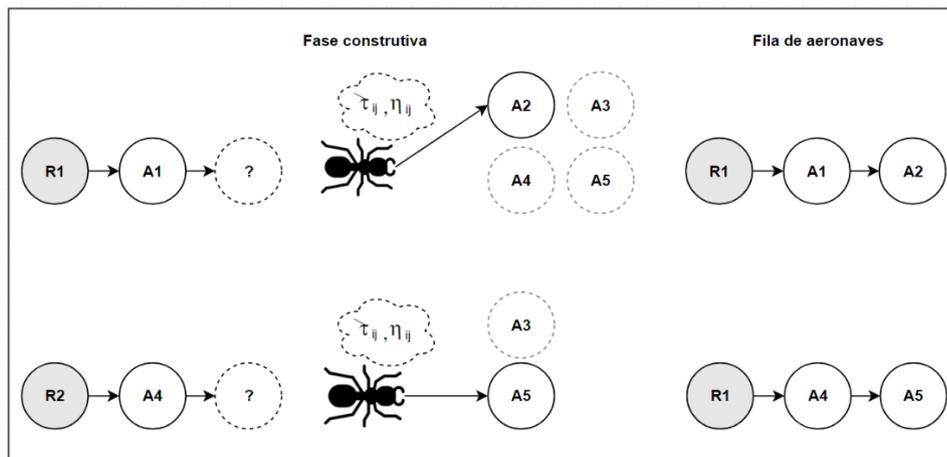
5.5 Representação da matriz heurística

A matriz de informação heurística da colônia é definida de forma análoga a matriz de feromônios definida na subseção anterior. O par (R_i, A_j) representa o quão conveniente é, para uma formiga da colônia, agendar a aeronave A_j na pista R_i levando em consideração a prioridade da aeronave e os custos de penalidade associados ao tempo de pouso pré-estipulado.

5.6 Construção da solução

Foi definida uma lista global de candidatos contendo os índices de todas as aeronaves. Antes de começar a construir a solução, uma formiga deve gerar sua própria lista de candidatos locais que corresponda às aeronaves disponíveis que apareceram no radar de alcance. Ela inicia seu caminho a partir do vértice D ; primeiro escolhe a pista onde o próximo avião vai pousar, essa escolha depende da carga da pista, ou qual pista estará livre o mais rápido possível.

Figura 12 – Construção de uma solução pela formiga artificial.



Fonte: O autor.

Conforme ilustrado na Figura 12, após selecionar uma pista, a formiga deve escolher a próxima aeronave a pousar naquela pista de sua lista de candidatos locais; essa escolha depende basicamente da prioridade de uma aeronave sobre a outra e da memória da colônia de formigas. A formiga deverá computar todos os tempos de pousos das aeronaves candidatas, bem como seus custos associados. A partir disso, uma aeronave é selecionada com base no cálculo da probabilidade cujo considera a influência do feromônio e a informação heurística; convém ressaltar que ao selecionar uma aeronave, ela é permanentemente removida da lista de candidatos da formiga (BENCHEIKH *et al.*, 2016).

5.7 Seleção da pista de pouso

A seleção da pista de pouso é descrita na equação (24). P_{Dr}^k representa a probabilidade de a formiga partir do vértice D e selecionar a pista r .

$$P_{Dr}^k = \begin{cases} \operatorname{argmin}_{r=1,\dots,R}(\text{QDTE de aviões pousam pista } r), & q < q_0 \\ r_0, & \text{caso contrário} \end{cases} \quad (24)$$

Onde:

- D representa o nó de início do grafo;
- $0 < q_0 < 1$ é uma constante do algoritmo que garante a diversificação;
- q É um valor tomado aleatoriamente sob o intervalo $[0, 1]$;
- r_0 É o índice de uma pista escolhido aleatoriamente em $\{1, \dots, R\}$.

No entanto, utilizar o número de aeronaves nas pistas não é uma boa estratégia visto que não garante que o intervalo de segurança (tempos de separação) das aeronaves está sendo respeitado.

A regra de probabilidade de (24) é válida somente para os casos em que todas as aeronaves possuem o mesmo tempo de separação.

Considerando o mais próximo da realidade, se faz necessário reformular a regra para selecionar as pistas de pouso de acordo com a sua disponibilidade de tempo para receber uma nova aeronave, assim tem-se a equação (25).

$$P_{Dr}^k = \begin{cases} \operatorname{argmin}_{r=1,\dots,R}(\min_{j \in I_k}(x_{i_0}^r + S_{i_0j})), & q < q_0 \\ r_0, & \text{caso contrário} \end{cases} \quad (25)$$

Onde:

- l_k representa a lista de aviões da formiga k que estão em espera para pousar;
- i_0 representa a última aeronave agendada para a pista r ;
- $x_{i_0}^r$ refere-se ao tempo de pouso da última aeronave agendada para a pista r pela formiga k ;
- S_{i_0j} refere-se ao tempo de separação entre o avião i_0 e j (onde i_0 pousa antes de j).

5.8 Informação Heurística

Uma vez selecionada a pista, a formiga deverá escolher uma aeronave para pousar. A informação heurística a ser definida depende da influência de dois fatores: (i) a prioridade da aeronave j ; (ii) o custo de penalidade associado a aeronave j . Esta influência é definida em (26).

De acordo com Bencheikh *et al.* (2016) a prioridade da aeronave pode ser definida utilizando os valores de primeiro pouso tempo, o tempo alvo, o custo de penalidade da aeronave ou até mesmo uma combinação dessas variáveis. A prioridade se refere ao custo atrelado à aeronave após ter um tempo de pouso computado. Como a formiga adiciona uma aeronave por ciclo do grafo, há a necessidade de que os tempos das aeronaves sejam computados na escolha das próximas aeronaves, em respeito a restrição de separação de segurança, visto em (17).

Dessa forma, fica evidente a característica dinâmica da informação heurística, demonstrando ser dependente das soluções parciais das formigas ao longo da iteração. Ou seja, à medida que a ordem de pouso vai sendo construída, o custo de penalidade de cada aeronave candidata precisa ser recalculado.

$$n_{rj} = \left(\frac{1}{(\text{Prioridade}(j) + 1)} \right)^{\beta_1} \cdot \left(\frac{1}{(\text{Penalidade}(j) + 1)} \right)^{\beta_2} \quad (26)$$

Onde:

- $Prioridade(j)$ é a prioridade da aeronave j ;
- $Penalidade(j)$ é o custo de penalidade da aeronave j ;
- β_1 e β_2 são os coeficientes de influência dos fatores prioridade e penalidade, respectivamente.

5.9 Seleção da aeronave

De acordo com a equação 27, $p_{rj}^k(t)$ representa a probabilidade da formiga k , na iteração t , estando na pista r , escolher o avião j para pousar.

$$p_{rj}^k(t) = \begin{cases} \frac{[\tau_{rj}]^\alpha [n_{rj}]^{(\beta_1 + \beta_2)}}{\sum_{l \in l} [\tau_{rl}]^\alpha [n_{rl}]^{(\beta_1 + \beta_2)}}, & \text{Se } j \in l \\ 0, & \text{caso contrário} \end{cases} \quad (27)$$

Onde:

- τ_{ij} define a quantidade de feromônio na trilha;
- n_{ij} define um valor heurístico conforme definido na seção 5.5;
- α é um parâmetro que determinam a influência da trilha de feromônio;
- $(\beta_1 + \beta_2)$ influência da informação heurística;
- l define o conjunto de possíveis soluções;
- l é o conjunto de nós de vizinhança viável da aeronave j , associado a k – ésima formiga, ou seja, o conjunto de aeronaves que a formiga ainda não atribuiu um tempo de pouso. A probabilidade de uma formiga escolher um nó que não pertence a l é zero.

5.10 Atribuição do tempo de pouso

A seguir, é descrita a expressão para atribuição do tempo de pouso em uma pista. A fim de gerar soluções viáveis, é necessário que as restrições definidas no Capítulo 3 sejam respeitadas. Portanto, o tempo de pouso atribuído deverá estar definido entre a janela de pouso $[E_i, L_i]$ bem como o intervalo de segurança (matriz de tempos de separação). Para esse cenário, podem ser considerados duas heurísticas que serão discutidas logo abaixo. Para ambos os casos, O é o conjunto de aviões que tiveram seus tempos de pouso atribuídos.

A primeira heurística atribui um tempo de pouso alvo se atender aos intervalos de segurança entre os aviões, caso contrário, atribui o primeiro horário para atender aos intervalos seguros. Isto é, a heurística prioriza atribuir o tempo de pouso ao tempo alvo pré-definido (que está contido na janela de pouso) sempre que possível, como consequência, reduzindo os custos. A equação (28) descreve o comportamento pretendido.

$$x_j = \max(t_{aj}, \max_{i \in O}(x_i + S_{ij})) \quad (28)$$

A partir de (29), observa-se que a segunda heurística se preocupa somente com os intervalos de segurança. Nesse cenário, a heurística prioriza atribuir o tempo de pouso mínimo à aeronave, refletindo na atribuição das penalidades do adiamento ocasionado. Tal abordagem tem como intuito reduzir os tempos de pouso das últimas aeronaves a entrarem na fila.

$$x_j = \max(e_j, \max_{i \in O}(x_i + S_{ij})) \quad (29)$$

Convém ressaltar que as duas heurísticas respeitam os itens definidos nos critérios de validação do algoritmo; a sua escolha dependerá da característica da instância em utilização; em um cenário, pode ser conveniente anular os custos de

pouso das aeronaves à medida que vão tendo seus tempos de pouso agendados nas pistas, o que pode acarretar custos de penalidade às últimas aeronaves. Em outro contexto, pode-se ter como objetivo reduzir as penalidades aplicadas às últimas aeronaves escalonadas, reduzindo o tempo total de espera dos aviões (BENCHEIKH *et al.*, 2016).

5.11 Atualização da matriz de feromônio

A equação (30) determina como os mecanismos de intensificação e diversificação foram adaptados para o ALP.

$$\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad \forall i \in A; \rho \in [0,1] \quad (30)$$

Onde:

- ρ é o coeficiente de evaporação ($\rho < 1$). É utilizado para evitar a convergência de soluções em um espaço de busca local;
- $\tau_{ij}(t + 1)$ é o valor do feromônio atualizado na aresta (i, j) ao término da iteração. Representando a escolha da aeronave j na pista i ;
- $\Delta\tau_{ij}(t)$ é a quantidade de feromônio depositada na aresta (i, j) pela colônia ao término da iteração, dada por (31). Representando a escolha da aeronave j na pista i .

$$\Delta\tau_{ij}(t) = \begin{cases} Q \\ C \end{cases} \quad \text{Se } j \in \text{Melhor solução} \quad (31)$$

Onde:

- Q é uma constante de atualização;
- C é o custo de penalidade total na aresta (i, j) ao término da iteração t ;
- *Melhor solução* é o caminho que possui o menor custo de penalidade total.

Estratégia de intensificação

O feromônio depositado nas trilhas construídas pelas formigas evapora ao longo do tempo, trata-se de uma característica natural de uma colônia de formigas. No entanto, a forma como o feromônio é depositado dependerá da qualidade da solução encontrada, assim como fazem algumas espécies de formigas da colônia no mundo real (DORIGO; STÜTZLE, 2004). Como critério de qualidade para criação das trilhas de feromônios, devido a limitação de recursos computacionais, a presente implementação intensificará apenas o caminho da solução de menor custo da iteração de modo a aumentar a convergência das soluções; todavia, é possível que todas as soluções encontradas contribuíssem depositando feromônio de forma proporcional a qualidade da solução.

Supondo um cenário com uma única pista e 6 aeronaves para pouso, a Tabela 5 ilustra 3 possíveis caminhos de soluções construídos por 3 formigas durante uma dada iteração. A sequência de pouso especificada pode ser compreendida como o conjunto de conexões entre as arestas, que é criada por cada formiga ao percorrer o grafo.

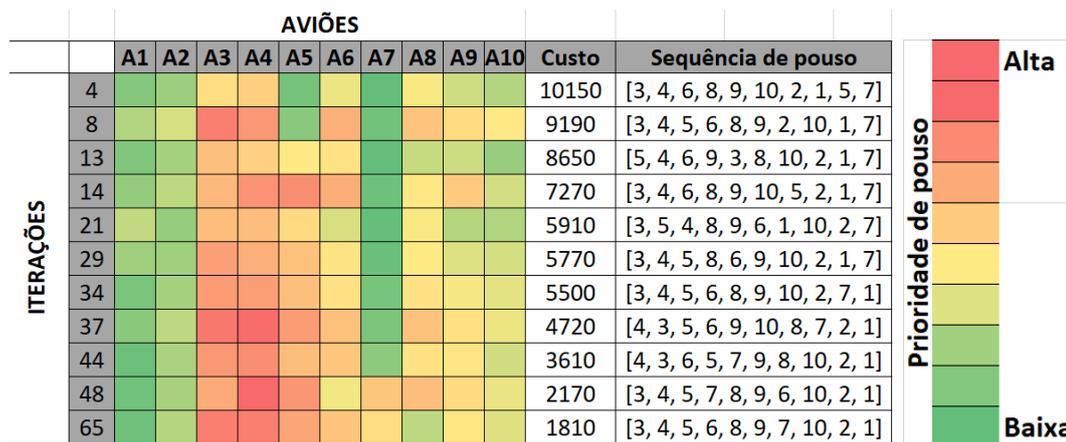
Tabela 5 – Exemplo da estratégia de intensificação.

Formiga	Sequência de pouso	Custo	Intensificar
F1	$\{(1, 4), (4, 2), (2, 3), (3, 6), (6, 5)\}$	1000	Sim
F2	$\{(1, 2), (2, 3), (3, 4), (4, 5), (5, 6)\}$	3550	Não
F3	$\{(1, 4), (4, 5), (5, 3), (3, 6), (6, 2)\}$	2520	Não

Fonte: O autor.

Ou seja, a ordem de pouso da formiga F1 terão os valores de feromônios de suas arestas intensificados de forma ponderada visto que foi a formiga que apresentou o menor custo, indicando ser uma solução de maior qualidade na iteração em questão.

Figura 13 – Quantidade de feromônio depositada por *slot*.



Fonte: O autor.

Percebe-se que a estratégia adotada deve considerar que a ordem de assimilação para a construção da solução tem fundamental importância para depósito do feromônio. Conforme mostrado na Figura 13, as primeiras aeronaves a entrarem na fila de pouso receberão maiores doses de feromônio. Para que isso seja possível, foi definida uma taxa percentual que é reduzida à medida que avança nas posições atribuídas pela formiga (primeira aeronave, segunda aeronave e assim por diante). Isso garante que as primeiras aeronaves receberam uma dose maior de feromônio, enquanto as sucessivas conexões receberão doses proporcionais. Tal estratégia tem como intuito auxiliar na escolha da ordem de pouso das formigas na próxima iteração, atuando na redução dos custos das soluções. (BENCHEIKH, 2011).

5.12 Pseudocódigo adaptado para o ALP

A Figura 14 mostra o pseudocódigo da meta-heurística colônia de formiga, o qual pode ser aplicada para solucionar problemas de otimização.

Figura 14 – Pseudocódigo do ACO aplicado ao ALP.

algoritmo ACO aplicado ao ALP

1. Inicializar os parâmetros (τ , n , colônia);
2. Para cada formiga da colônia:
 - 2.1. Inicializar lista de aeronaves candidatas;
 - Enquanto** possuir aeronaves candidatas:
 - 2.2. Selecionar pista de acordo com equação (25);
 - 2.3. Selecionar aeronave de acordo com equação (27);
 - 2.4. Atualizar informação heurística;
 - 2.5. Adicionar aeronave na lista de soluções para a pista a selecionada, remover a aeronave da lista de candidatas;
 - 2.6. Retornar ao início do grafo até que a lista de candidatas esteja vazia.
 - Fim do enquanto**
3. Atualizar feromônio de acordo com equação (30);
4. Repetir passos 2, 3 e 4 até atingir condição de parada.

Fonte: Baseado em Bencheikh *et al.* (2011).

6 Experimento

Inicialmente foram realizados estudos a respeito do estado da arte dos algoritmos de colônia de formigas aplicados ao problema do ALP com foco em entender sobre a implementação e suas variações através dos trabalhos de Beasley *et al.* (2001), Bencheikh *et al.* (2011) e Zhan *et al.* (2010), assim como, também, verificar outras formas de melhoria do processo de diversificação e representação do algoritmo a partir dos trabalhos de Colorni *et al.* (1994), Bencheikh *et al.* (2016) e IKLI *et al.* (2019).

Após a análise do problema, foi implementado o algoritmo baseado na meta-heurística Otimização por Colônia de Formigas descrita no Capítulo 4, tendo em vista a obtenção de uma solução válida para o problema já anteriormente descrito. Devido a limitação de recursos computacionais, optou-se por uma implementação estritamente sequencial. Portanto, dada a representação por grafo, ilustrada na Figura 10, em cada iteração, apenas uma formiga percorrerá o grafo por vez. Uma implementação paralela permitirá que todas as formigas saíssem do nó D ao mesmo tempo, construindo suas soluções de forma paralela.

O algoritmo foi implementado utilizando a linguagem de programação *Python*, na versão 3.9. Foram utilizadas bibliotecas adicionais como *NumPy*, *Pandas* e *Networkx*. A implementação tem caráter *open-source*, sendo disponibilizado sob a licença do *Massachusetts Institute of Technology* (MIT) em um repositório público do autor.

6.1 Descrição das instâncias utilizadas

A verificação do algoritmo será dada por meio do uso de dados teóricos disponibilizados na *OR-Library* o qual se caracteriza como um conjunto de base de dados para uma variedade de problemas de pesquisa operacional. Foi descrita originalmente por Beasley (1990).

O trabalho utilizou o conjunto de dados teóricos para o Problema de Pouso de Aeronaves (13 conjuntos de instâncias) que se intitulam com o prefixo *airland*, o qual

também foram utilizados por Beasley *et al.* (2000). A Tabela 6 agrupa as instâncias pelo tamanho de aeronaves para pouso.

Tabela 6 – Tamanho das instâncias disponíveis.

Tamanho	Instância(s)
10	<i>airland_1</i>
20	<i>airland_2, airland_3, airland_4, airland_5</i>
30	<i>airland_6</i>
44	<i>airland_7</i>
50	<i>airland_8</i>
100	<i>airland_9</i>
150	<i>airland_10</i>
200	<i>airland_11</i>
250	<i>airland_12</i>
500	<i>airland_13</i>

Fonte: Baseado em Beasley *et al.* (2000).

Para explicar os metadados comum a todas as instâncias, será considerada a instância *airland_1*, que possui 10 aeronaves. A Tabela 7 descreve que para cada aeronave A_i , existe uma janela de pouso $[E_i, L_i]$, um tempo alvo para pouso Ta_i os respectivos custos de penalidade unitários para pousos antecipados C_i^- e tardios C_i^+ . Tomando a primeira linha da Tabela 7, a aeronave A_1 possui um tempo mínimo de pouso E_i de 129, um tempo alvo Ta_i de 155 e um tempo máximo L_i permitido de 599.

Tabela 7 – Tempos das aeronaves da instância *airland_1*.

A_i	E_i	Ta_i	L_i	C_i^-	C_i^+
01	129	155	599	10	10
02	195	258	744	10	10
03	89	98	510	30	30
04	96	106	521	30	30
05	110	123	555	30	30
06	120	135	576	30	30
07	124	138	577	30	30
08	126	140	573	30	30
09	135	150	591	30	30
10	160	180	657	30	30

Fonte: Baseado em Beasley *et al.* (2000).

A Tabela 8 descreve os tempos de separação mínimos S_{ij} , dado em unidades de tempo, entre cada um dos pares de aeronaves da instância *airland_1*. Em outras palavras, cada aeronave possui uma lista dos tempos de separação mínimos para que as outras aeronaves possam utilizá-los, garantindo a segurança do sequenciamento.

Em um cenário onde a aeronave A_1 pousa antes da aeronave A_2 , é sabido, por meio da restrição (17), que o tempo de pouso da aeronave A_2 precisa ser acrescido com o tempo de separação requisitado pela aeronave A_1 . Analisando a Tabela 8, este tempo é de 3 unidades de tempo. Além disso, ainda tomando aeronave A_1 como exemplo, em observância a restrição (14), que garante que uma aeronave pousa somente em uma única pista, a instância aplica um tempo de separação de 9999 unidades de tempo para esses casos; na prática, o valor é inalcançável se comparado aos demais tempos, impedindo que a restrição seja violada.

Tabela 8 – Tempo de separação entre as aeronaves do exemplo 01

A_i	1	2	3	4	5	6	7	8	9	10
01	9999	3	15	15	15	15	15	15	15	15
02	3	9999	15	15	15	15	15	15	15	15
03	15	15	9999	8	8	8	8	8	8	8
04	15	15	8	9999	8	8	8	8	8	8
05	15	15	8	8	9999	8	8	8	8	8
06	15	15	8	8	8	9999	8	8	8	8
07	15	15	8	8	8	8	9999	8	8	8
08	15	15	8	8	8	8	8	9999	8	8
09	15	15	8	8	8	8	8	8	9999	8
10	15	15	8	8	8	8	8	8	8	9999

Fonte: Baseado em Beasley *et al.* (2000)

6.2 Descrição do experimento

Foram selecionadas as instâncias de número 1, 2, 3, 4, 5 e 8 disponibilizados na *OR-Library*. O presente trabalho assume que o tempo de separação entre dois aviões em pistas distintas será 0. Foram executados 3 ciclos de experimentos, cada um com uma configuração dos parâmetros pré-definida, aplicadas igualmente para todas as instâncias. Foi utilizada a seguinte configuração processador Intel Core i5-10210U 1.60GHz em conjunto com 8GB RAM. O Windows 10 foi o sistema operacional utilizado para execução do algoritmo. A Tabela 9 mostra os parâmetros de execução utilizados em cada ciclo: o tamanho da colônia, influência do feromônio, influência da informação heurística e a taxa de evaporação, o qual foram selecionados de forma experimental. Cada instância será executada nos cenários única e múltiplas pistas. Os resultados encontrados foram registrados em uma planilha eletrônica (ciclo, instância, quantidade de pistas, custo e tempo de processamento).

Tabela 9 – Parâmetros utilizados por ciclo de experimento.

Ciclo	Iterações	Colônia	α	β_1	β_2	ρ
1	100	75	2	6	5	0,5
2	50	50	1	2	4	0,9
3	400	tamanho da instância	3	2	0,5	0,1

Fonte: O autor.

7 Resultados

Neste capítulo serão discutidos os resultados encontrados em cada ciclo do experimento. Os parâmetros utilizados em cada ciclo, as soluções encontradas e seus tempos computacionais serão analisados para que haja uma melhor compreensão da problemática. Além disso, tais resultados são comparados com aqueles obtidos em Bencheikh et al. (2011) e também com os experimentos recentes realizados por Silva (2022), que utiliza algoritmo genético (do inglês, *Genetic Algorithm* - GA) para uma formulação do problema equivalente àquela apresentada no Capítulo 3, considerando o cenário de uma única pista.

7.1 Resultados encontrados

A Tabela 10 mostra as instâncias selecionadas, a quantidade de pistas disponíveis para agendamento das aeronaves (coluna R), os custos e o tempos de execução de cada um dos três ciclos de experimentos. Os resultados foram agrupados por ciclo de experimento. A coluna ACA (do inglês, *Ant Colony Algorithm*) apresenta os resultados obtidos pela implementação paralela do pseudocódigo da Figura 11 realizada por Bencheikh et al. (2011).

Tabela 10 – Comparação dos resultados por ciclo de experimento.

Instância	R	Ciclo 1		Ciclo 2		Ciclo 3		ACA	
		Custo	Tempo	Custo	Tempo	Custo	Tempo	Custo	Tempo
airland_1	1	2650	25,79"	1750	6,97"	1150	10,45"	1150	0,49"
	2	300	22,72"	120	6,59"	280	9,65"	120	0,49"
	3	0	18,60"	0	6,72"	150	11,37"	0	0,43"
airland_2	1	34210	42,66"	29290	14,20"	1720	33,65"	1840	0,93"
	2	4130	43,51"	1120	13,65"	300	32,79"	210	0,82"
	3	120	44,07"	1230	13,91"	450	31,24"	0	0,71"
airland_3	1	63220	1' 37"	45210	28,72"	1650	1' 28"	2540	1,42"
	2	25390	1' 26"	22310	28,72"	660	1' 26"	60	1,31"
	3	3050	1' 22"	5490	28,78"	0	1' 34"	0	1,20"
airland_4	1	66400	1' 32"	55610	28,68"	4480	2' 21"	4820	1,42"
	2	17400	1' 25"	23290	25,69"	1010	1' 50"	680	1,26"
	3	3560	1' 25"	7070	25,50"	660	1' 57"	130	1,20"
	4	480	1' 24"	2980	29,34"	390	1' 22"	0	1,09"
airland_5	1	83430	1' 46"	59680	30,48"	4800	1' 29"	6260	1,42"
	2	30290	1' 41"	20990	30,25"	1390	1' 26"	1210	1,31"
	3	10370	1' 32"	7730	29,87"	630	2' 50"	330	1,2"
	4	750	1' 44"	5920	28,79"	420	3' 09"	0	1,09"
airland_8	1	468750	20' 47"	371995	6' 40"	13220	53' 27"	13840	10,01"
	2	308090	19' 48"	299480	6' 14"	10745	51'	835	9,72"

Fonte: O autor.

7.2 Análise dos resultados

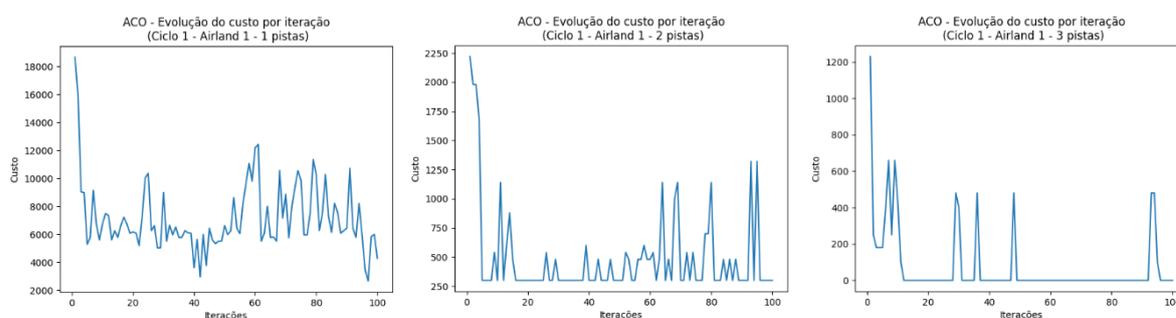
Os experimentos demonstraram a importância do feromônio, da informação heurística, do tamanho da colônia e da quantidade de iterações para resolução das instâncias. A combinação de seus valores foram fundamentais para atuar na convergência da função objetivo e na eficiência do algoritmo. Os parâmetros foram definidos experimentalmente, demonstrando ser uma abordagem eficaz para escolha de uma configuração capaz de prover soluções viáveis com baixo custo, próximas da implementação proposta por Bencheikh *et al.* (2011), conforme ilustra a Tabela 10.

No Ciclo 1, a configuração escolhida teve como intuito promover a diversificação da quantidade de soluções geradas por iteração, principalmente por meio do tamanho da colônia e da influência do feromônio associada à determinação

de que 50% do feromônio computado da iteração anterior deverá ser mantido, para visualizar o efeito, foi estipulada uma quantidade de 100 iterações. Além disso, será avaliada a capacidade das formigas construírem soluções que priorizam as aeronaves com maior prioridade de pouso ($\beta_1 > \beta_2$).

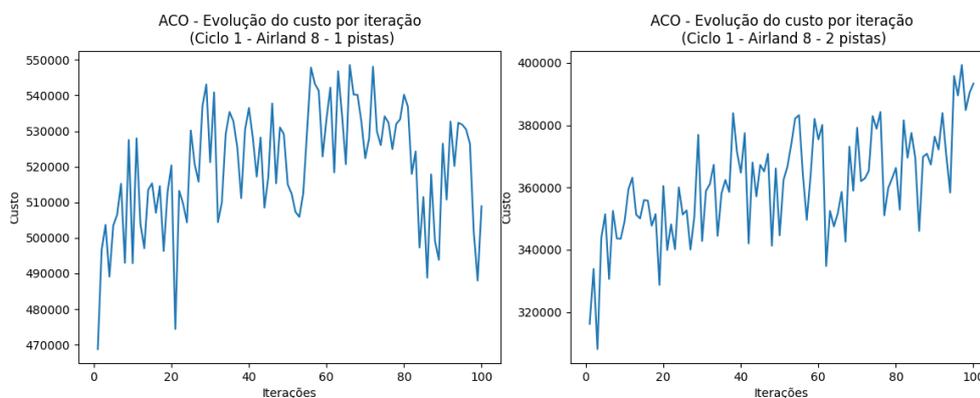
Os resultados encontrados demonstraram ser elevados, no entanto, foi percebida a influência do feromônio ao longo das iterações, reduzindo gradativamente os custos computados. No entanto, a colônia de formigas apresentou um comportamento de estagnação, foram observados picos nos custos durante as iterações, demonstrando que a taxa de evaporação escolhida se manteve elevada, fazendo com que a colônia reconstruísse as soluções. Nesse sentido, a colônia tendeu a ficar presa em mínimos locais mesmo com as estratégias de diversificação possíveis, conforme ilustra os resultados obtidos para a instâncias *airland_1* e *airland_8* que podem ser vistos na Figura 15 e Figura 16, respectivamente. A combinação do tamanho da colônia de formigas e da quantidade de iterações produziram elevados tempos de computação. De um modo geral, as soluções estavam distantes dos valores nominais encontrados por Bencheikh *et al.* (2011).

Figura 15 – Evolução dos custos da instância *airland_1* no Ciclo 1.



Fonte: O autor.

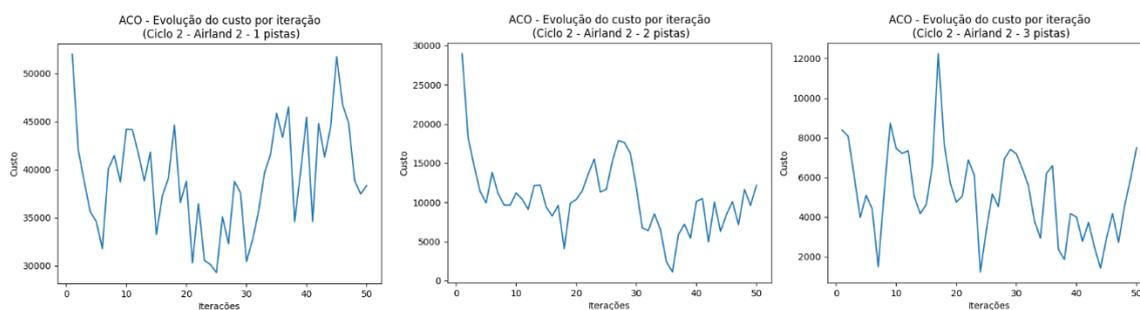
Figura 16 – Comparativo da evolução dos custos para airland_8 no Ciclo 1.



Fonte: O autor.

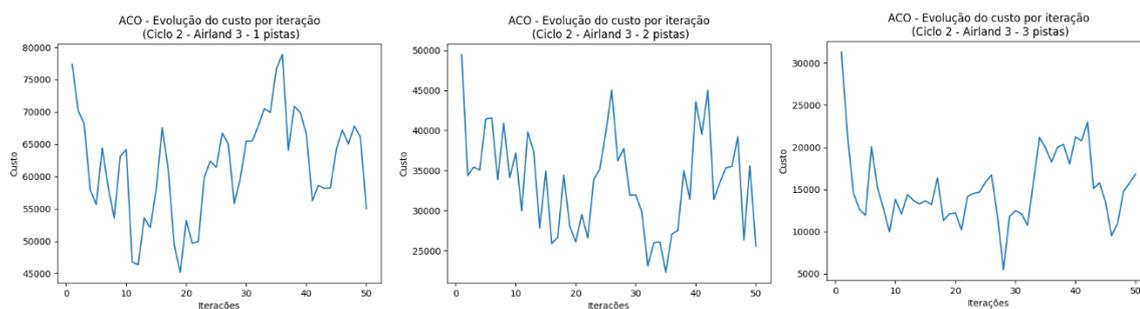
O Ciclo 2 foi executado reduzindo-se a quantidade de iterações e formigas, além de experimentar a qualidade das soluções de uma colônia que prioriza as informações locais como a prioridade de pouso e custo computado, com ênfase para este último ($\beta_2 > \beta_1$), visando obter a sequência com os menores custos individuais possíveis. Os ajustes de parâmetros definidos para a execução do Ciclo 2 tinham como objetivo a redução do tempo de processamento e uma redução no comportamento de estagnação observado no ciclo anterior. Para isso, também foi reduzida a influência do conhecimento da colônia de formigas para a definição da ordem de pouso. Para intensificar a hipótese, somente 10% do feromônio continuava presente ao término de cada iteração além de reduzir a quantidade de iterações pela metade.

Os resultados encontrados obtiveram uma redução total média nos custos para ambos os casos, mantendo-se mais próximos dos resultados computados por Bencheikh *et al.* (2011). Para o cenário com única pista, foi percebida uma redução do custo total médio de 21% em relação ao ciclo anterior.

Figura 17 – Evolução dos custos da instância *airland_2* no Ciclo 2.

Fonte: O autor.

Para o cenário com múltiplas pistas, houve uma redução do custo total médio próximo a 1.5%. O comportamento de estagnação foi amortecido, no entanto, as soluções apresentaram baixa convergência ao longo das iterações, onde a colônia apresentou alto grau de diversificação, mas baixa intensificação, tal conclusão poder ser inferida por meio da análise da evolução dos custos para as instâncias *airland_2* (Figura 17) e *airland_3* (Figura 18) nos cenários de única e múltiplas pistas.

Figura 18 – Evolução dos custos da instância *airland_3* no Ciclo 2.

Fonte: O autor.

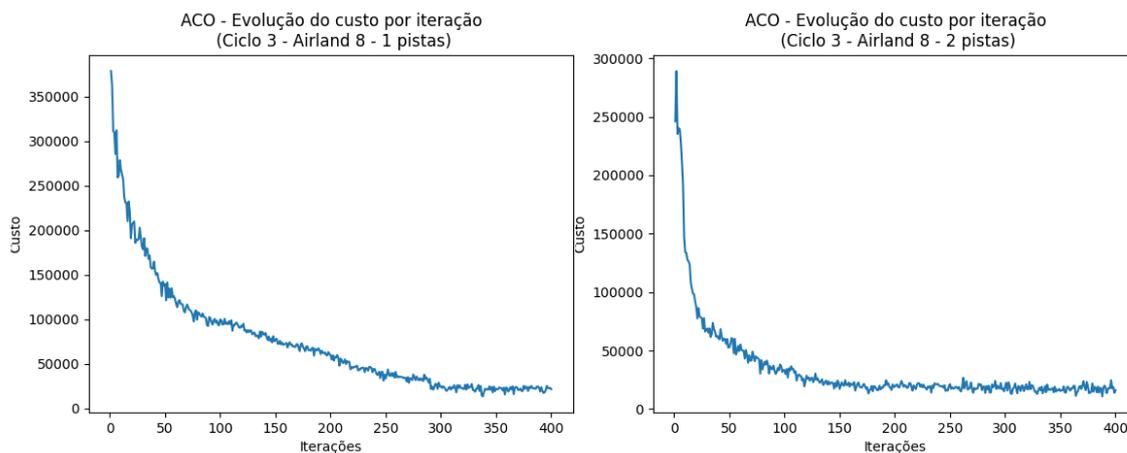
Sendo assim, ficou evidente a redução do mecanismo de intensificação, por meio do uso da inteligência coletiva, o qual reduziu o comportamento de progressão observado no ciclo anterior, principalmente para o cenário com múltiplas pistas onde o custo sofreu aumento na execução das instâncias *airland_3*, *airland_4*, e *airland_4*

; diferentemente do cenário com única pista, que apresentou expressiva redução nos custos optando pela predominância das escolhas locais em detrimento do conhecimento coletivo. Por fim, foi observada uma redução do tempo médio de processamento, em torno de 68%.

Por fim, para o Ciclo 3, a configuração dos parâmetros da informação heurística visou intensificar a influência da escolha das formigas na priorização de aeronaves com maior prioridade de pouso ($\beta_1 > \beta_2$), associando ao fato que é mantido cerca de 90% do feromônio depositado pelas formigas na iteração anterior. Além disso, percebe-se que aumentar a quantidade de iterações não trouxe ganhos significativos a solução comparada ao ciclo anterior, após encontrar o menor custo possível, o algoritmo apresentou um comportamento de estagnação, mantendo-se próxima a melhor solução computada. Os testes revelaram que tal configuração conseguiu distribuir os custos de penalidade de forma mais uniforme, reduzindo o custo de penalidade total. Para este ciclo, foi registrado acréscimo no tempo de computação em torno de 8%, se comparado ao ciclo anterior. No entanto, os resultados encontrados tiveram uma redução no custo médio em torno de 91% se comparado ao Ciclo 2 e de 95% se comparado ao Ciclo 1.

A configuração dos parâmetros foi capaz de reduzir os custos totais para ambos os cenários (única e múltiplas pistas), demonstrando que as formigas fizeram o melhor uso possível das estratégias de diversificação e intensificação ao longo das iterações. Além disso, percebe-se que aumentar a quantidade de iterações não trouxe ganhos significativos para as instâncias com até 20 aeronaves, onde a colônia convergiu durante as primeiras iterações; as demais iterações foram marcadas por um comportamento de estagnação, mantendo-se próxima a melhor solução computada. Contudo, para a instância *airland_8*, foi notória a influência da inteligência coletiva durante as iterações, trazendo gradativas reduções ao longo de todas as iterações, a Figura 16 destaca a evolução dos custos para esta instância para o primeiro ciclo enquanto que a Figura 19 para trás os resultados o último ciclo.

Figura 19 – Comparativo da evolução dos custos para airland_8 no Ciclo 3.



Fonte: O autor.

Conclui-se que se faz necessário um balanceamento entre a qualidade das soluções e os custos computacionais associados. A combinação do aumento da influência heurística das formigas para priorizarem aeronaves mais prioritárias em conjunto com a redução da taxa de evaporação da colônia de formigas contribuiu para intensificar a redução dos custos no espaço de busca ao longo das iterações, ainda garantindo a capacidade para exploração de soluções de maiores custos. Além disso, a presente implementação em sua versão sequencial requer consideráveis tempos de processamento, demonstrando a sua natureza paralela, onde as formigas da colônia podem construir suas soluções de forma autônoma e em conjunto.

A presente implementação da Colônia de Formigas esbarrou na limitação de somente uma formiga poder percorrer o grafo por vez. No entanto, o uso de uma meta-heurística construtiva modelada sobre uma estrutura de grafos tem a principal vantagem de gerar soluções estritamente viáveis, não necessitando esforços adicionais para avaliação da viabilidade das soluções, o que pode ser benéfico para instâncias do mundo real. Além disso, não foram percebidas diferenças significativas nos tempos computados entre os cenários de única e múltiplas pistas, possibilitando

que todas as pistas disponíveis em um dado aeroporto sejam utilizadas na melhor utilização dos financeiros e dos custos operacionais encontrados.

O ACO se sobressai para os casos onde é possível estabelecer marcos expressivos entre as prioridades das aeronaves, sendo capaz de dividir os custos uniformemente entre elas. Estes marcos não são possíveis de serem determinados facilmente quando o algoritmo esbarra com janelas de pouso conflitantes ou com tempos de separação muito próximos entre as aeronaves, provocando a colônia a punir as aeronaves que aterrissam ao final da fila, devido a sua estratégia de intensificação gulosa o qual tenta buscar os menores custos durante a composição das soluções.

Além disso, foi observada a alta sensibilidade dos parâmetros, destacando a sua capacidade de se ajustar ao problema em questão onde pequenas modificações nos parâmetros foram capazes de reduzir as diferenças nos custos ao longo dos ciclos. Portanto, conclui-se que diferentes configurações nos parâmetros do ACO podem influenciar instâncias distintas, seja nos cenários com única e múltiplas pistas, demonstrando a escalabilidade do algoritmo para aplicações em cenários do mundo real.

7.3 Comparativo entre ACO e GA

Em Silva (2022) é apresentado um experimento utilizando algoritmo genético para a formulação do problema aqui apresentada, considerando o cenário de uma única pista.

A seguir será apresentado um comparativo entre os experimentos realizados neste trabalho usando ACO e os experimentos apresentados por Silva (2022) usando GA, para o cenário de uma única pista.

Silva (2022) realizou três ciclos de experimentos utilizando diferentes parâmetros. A Tabela 10 exibe as instâncias consideradas, seus tamanhos, o algoritmo utilizado, o custo obtido e o tempo de processamento resultando do experimento com o ACO e o GA. Cada linha ACO apresenta o resultado obtido pelo algoritmo implementado pelo presente trabalho e cada linha GA apresenta os resultados do Algoritmo Genético implementado por Silva (2022). Enquanto o ACO

possui uma natureza construtiva, o Algoritmo Genético apresenta uma abordagem estocástica para encontro de soluções viáveis.

Tabela 11 – Comparativo de resultados entre ACO e GA.

Instância	Algoritmo	Ciclo 1		Ciclo 2		Ciclo 3	
		Custo	Tempo	Custo	Tempo	Custo	Tempo
airland_1	ACO	6070	8,86"	2170	24,74"	1150	22,02"
	GA	3080	1,10"	1730	3,79"	960	23,48"
airland_2	ACO	41110	19,22"	24980	1' 11"	1870	1' 14"
	GA	7280	1,84"	3120	6,02"	2260	39,01"
airland_3	ACO	64220	38,89"	54670	2'	2730	3' 20"
	GA	11280	2,89"	4490	8,58"	1410	58,38"
airland_4	ACO	78340	23,06"	52050	1' 48"	4550	1' 47"
	GA	15910	2,76"	10460	8,27"	6750	59,51"
airland_5	ACO	97150	26,06"	68270	2' 04"	5540	1' 36"
	GA	16110	3,07"	10990	8,30"	6730	65,44"
airland_8	ACO	468750	20' 47"	371995	6' 40"	13220	53' 27"
	GA	62335	13,05"	31290	36,97"	17725	230,87"

Fonte: O autor e SILVA (2022).

Conforme apontado por Silva (2022), a principal característica do Algoritmo Genético, em sua versão sequencial, reside na capacidade de geração de soluções viáveis em um curto espaço de tempo, mesmo fazendo validações adicionais quanto a viabilidade das soluções geradas pelos genes, visto que a implementação trabalha com um espaço de soluções viáveis e inviáveis – podendo aproveitar este último para a composição de soluções viáveis nas próximas gerações.

Por fim, por mais que os algoritmos apresentados tenham abordagens de intensificação e diversificação distintas, se mostraram capazes de promover soluções viáveis e próximas dos resultados teóricos encontrados na literatura. O último ciclo de experimento trouxe sutis diferenças entre os algoritmos, demonstrando que estratégias populacionais e construtivas são beneficiadas de acordo com as características particulares de cada instância.

8 Considerações Finais e Trabalhos Futuros

Futuros

A meta-heurística Otimização por Colônia de Formigas, proposta por Dorigo (1992), tem sido empregada para a resolução de problemas matemáticos, em especial aos problemas de otimização combinatória que lidam com o escalonamento de recursos finitos. O Problema do Pouso de Aeronaves tem como objetivo organizar um conjunto de aviões em um aeroporto, o qual possui uma quantidade limitada de pistas. Para que isso seja possível, aspectos como a precedência de pouso e os tempos de separação precisam ser respeitados.

Este trabalho apresentou um algoritmo de Colônia de Formigas para o Problema de Pouso de Aeronaves para o caso de múltiplas pistas baseando-se no pseudocódigo descrito por Bencheikh *et al.* (2011). A implementação criada foi capaz de prover soluções que se encontram em conformidade com as restrições da formulação matemática do problema, sendo uma alternativa aos métodos exatos para a resolução do problema combinatórios de tempo polinomial não determinístico.

Como contribuição à implementação, foi elaborada uma estratégia de intensificação baseada na ordem de prioridade dos *slots* das pistas, onde posições iniciais são consideradas mais prioritárias, recebendo uma maior taxa do feromônio, a abordagem foi capaz de auxiliar na convergência das soluções requisitando um menor número de formigas e iterações. Foram realizados três ciclos de experimentos, onde o algoritmo desenvolvido foi testado utilizando o conjunto de instâncias disponível na OR-Library, envolvendo 10 a 50 aeronaves e 1 a 4 pistas. Os resultados foram analisados e comparados em conjunto com as soluções obtidas pelo algoritmo genético implementado por SILVA (2022). Foi possível reduzir os custos e o tempo computacional envolvido mediante ajustes dos parâmetros da colônia de formigas. A implementação trouxe resultados satisfatórios para ambos os cenários (com única e múltiplas pistas) estando próximos, quando não iguais, aos resultados encontrados por Bencheikh *et al.* (2011). Conclui-se que as análises realizadas apontaram a notória robustez e aplicabilidade da meta-heurística ao gerar valores da função objetivo

próximos de outras meta-heurísticas aplicadas à problemática, como se deu comparativo com o Algoritmo Genético implementado por Silva (2022).

Foram encontrados maiores desafios na compreensão da formulação matemática, bem como nos esforços em pesquisas até a escolha do algoritmo apresentado, analisando a estrutura de dados que melhor se adequasse ao problema e algoritmo selecionado. Além da limitação de recursos computacionais para realização de experimentos com instâncias rentes à realidade.

Como possíveis trabalhos futuros, será possível utilizar o núcleo da implementação desenvolvida neste trabalho para fornecer subsídios para a concepção de uma versão multitarefas do algoritmo, onde todas as formigas seriam capazes de percorrer o grafo de forma concorrente. Além disso, também é possível aprimorar a estratégia de atribuição dos tempos de pouso, tornando-a capaz de reajustar os tempos de pouso de aeronaves, reduzindo o custo total, sendo necessária a adição de uma camada adicional para verificação da viabilidade das soluções. Por fim, realizar experimentos em instâncias com maior quantidade de aeronaves podem ajudar a propor outros mecanismos de busca que tornem o algoritmo mais compatível com a realidade do problema.

Referências

ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANESSE, H. **Pesquisa Operacional: Para Cursos de Engenharia**. 1ª Edição. Editora Campus-Elsevier, Rio de Janeiro – RJ, 2006.

AVELLA, Pasquale; MANNINO, Carlo; VASILYEV, Igor. **Time-indexed formulations for the runway scheduling problem**. *Transportation Science*, v. 51, n. 4, p. 1196-1209, 2017.

BARROS NETO, Benício de; Scarmínio, Ieda S. e Bruns, Roy E. **Como fazer experimentos: pesquisa e desenvolvimento na ciência e na indústria**. 2ª ed. Campinas, SP: Editora da UNICAMP, 2002.

BEASLEY, John E.; KRISHNAMOORTHY, Mohan; SHARAIHA, Yazid; ABRAMSON, David. **Scheduling aircraft landings—the static case**. *Transportation science*, v. 34, n. 2, p. 180-197, 2000.

BEASLEY, John E. **OR-Library: distributing test problems by electronic mail**. *Journal of the operational research society*, v. 41, n. 11, p. 1069-1072, 1990.

BEASLEY, John E.; SONANDER, Julia; HAVELOCK, P. **Scheduling aircraft landings at London Heathrow using a population heuristic**. *Journal of the operational Research Society*, v. 52, n. 5, p. 483-493, 2001.

BENCHEIKH, Ghizlane; BOUKACHOUR, Jaouad; ALAOUI, Ahmed EL Hilali. **Improved ant colony algorithm to solve the aircraft landing problem**. *International Journal of Computer Theory and Engineering*, v. 3, n. 2, p. 224, 2011.

BENCHEIKH, Ghizlane; BOUKACHOUR, Jaouad; ALAOUI, Ahmed El Hilali. **A memetic algorithm to solve the dynamic multiple runway aircraft landing problem**. *Journal of King Saud University-Computer and Information Sciences*, v. 28, n. 1, p. 98-109, 2016.

BALAKRISHNAN, Hamsa; CHANDRAN, Bala. **Scheduling aircraft landings under constrained position shifting**. In: *AIAA guidance, navigation, and control conference and exhibit*. 2006. p. 6320.

BLUM, Christian; ROLI, Andrea. **Metaheuristics in combinatorial optimization: Overview and conceptual comparison**. *ACM computing surveys (CSUR)*, v. 35, n. 3, p. 268-308, 2003.

COLORNI, A.; DORIGO, M.; MANIEZZO, V.; TRUBIAN, M. **Ant system for job-shop scheduling**. *JORBEL-Belgian Journal of Operations Research, Statistics, and Computer Science*, v. 34, n. 1, p. 39-53, 1994.

CUNHA, A. G.; TAKAHASHI, R.; ANTUNES, C. H. **Manual de Computação Evolutiva**. University of Chicago: Coimbra: Imprensa da Universidade de Coimbra, 2012.

DECEA: **Capacidade do Sistema de Pistas – MCA 100-14**. Disponível em <https://publicacoes.decea.mil.br/publicacao/mca-100-14>, 2015. Acesso em 03 de outubro de 2022.

DECEA: **Serviço de gerenciamento de fluxo de tráfego aéreo – ICA 100-22**. Disponível em <https://publicacoes.decea.mil.br/publicacao/ica-100-22>, 2018. Acesso em 17 de setembro de 2022.

DECEA: **Serviços de tráfego aéreo – ICA 100-37**. Disponível em <https://publicacoes.decea.mil.br/publicacao/ica-100-37>, 2020. Acesso em 17 de setembro de 2022.

DORIGO, M.; STÜTZLE, T. **Ant colony optimization**. Boston: MIT Press, 2004.

DORIGO, Marco; MANIEZZO, Vittorio; COLORNI, Alberto. **The ant system: An autocatalytic optimizing process**. 1991.

COLORNI, A.; DORIGO, M.; MANIEZZO, V. **An Investigation of some Properties of an " Ant Algorithm"**. In: *Ppsn*. 1992.

FURINI, F.; KIDD, M. P.; PERSIANI, C. A.; TOTH, P. **Improved rolling horizon approaches to the aircraft sequencing problem**. *Journal of Scheduling*, v. 18, n. 5, p. 435-447, 2015.

GHONIEM, Ahmed; SHERALI, Hanif D.; BAIK, Hojong. **Enhanced models for a mixed arrival-departure aircraft sequencing problem**. *INFORMS Journal on Computing*, v. 26, n. 3, p. 514-530, 2014.

GOSS, Simon; ARON, S.; DENEUBOURG, J. L.; PASTEELS, J. M. **Self-organized shortcuts in the Argentine ant**. *Naturwissenschaften*, v. 76, n. 12, p. 579-581, 1989.

IATA: **2036 Forecast Reveals Air Passengers Will Nearly Double to 7.8 Billion**. Disponível em <https://www.iata.org/en/pressroom/pr/2017-10-24-01>, 2017. Acesso em 15 de outubro de 2022.

IKLI, S.; MANCEL, C.; MONGEAU, M.; OLIVE, X.; RACHELSON, E. **An optimistic planning approach for the aircraft landing problem**. In: *ENRI International Workshop on ATM/CNS*. Springer, Singapore, 2019. p. 173-188.

IKLI, S.; MANCEL, C.; MONGEAU, M.; OLIVE, X.; RACHELSON, E. **The aircraft runway scheduling problem: A survey**. *Computers & Operations Research*, v. 132, p. 105336, 2021.

LEIGHFISHER (FIRM); LANDRUM; BROWN. **Evaluating Airfield Capacity**. Transportation Research Board, 2012.

SOYKAN, Bulent. **A hybrid Tabu/Scatter Search algorithm for simulation-based optimization of multi-objective runway operations scheduling**. Old Dominion University, 2016.

SILVA, Nilton Vieira. **Algoritmo de Otimização utilizando Heurística Genética Populacional para o Problema do Agendamento de Aeronaves em Única Pista**. Orientadora: Dra. Eliane Maria Loiola. 2022. TCC (Graduação) – Curso de Engenharia de Computação, Universidade de Pernambuco, 2022.

TRIVIZAS, Dionyssios A. **Optimal scheduling with maximum position shift (MPS) constraints: A runway scheduling application**. The Journal of Navigation, v. 51, n. 2, p. 250-266, 1998.

WEN, Min. **Algorithms of scheduling aircraft landing problem**. 2005. Dissertação de Mestrado. Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark.

ZHAN, Z. H.; ZHANG, J.; LI, Y.; LIU, O.; KWOK, S. K.; LP, W. H.; KAYNAK, O. **An efficient ant colony system based on receding horizon control for the aircraft arrival sequencing and scheduling problem**. IEEE Transactions on Intelligent Transportation Systems, v. 11, n. 2, p. 399-412, 2010.