

UNIVERSIDADE FEDERAL DE JUIZ DE FORA  
FACULDADE DE ENGENHARIA  
BACHARELADO EM ENGENHARIA COMPUTACIONAL

# Uma Abordagem Heurística para o Problema de Roteamento e Localização de Facilidades com o Uso de Drones

Davi Rezende Domingues

JUIZ DE FORA  
JANEIRO, 2023

# Uma Abordagem Heurística para o Problema de Roteamento e Localização de Facilidades com o Uso de Drones

DAVI REZENDE DOMINGUES

UNIVERSIDADE FEDERAL DE JUIZ DE FORA  
FACULDADE DE ENGENHARIA  
INSTITUTO DE CIÊNCIAS EXATAS  
Bacharelado em ENGENHARIA COMPUTACIONAL

Orientador: Luciana Brugiolo Gonçalves

JUIZ DE FORA  
JANEIRO, 2023

# UMA ABORDAGEM HEURÍSTICA PARA O PROBLEMA DE ROTEAMENTO E LOCALIZAÇÃO DE FACILIDADES COM O USO DE DRONES

Davi Rezende Domingues

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DA FACULDADE DE ENGENHARIA DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM ENGENHARIA COMPUTACIONAL.

Aprovada por:

Luciana Brugiolo Gonçalves  
Doutora em Ciência da Computação DCC - UFJF

Carlos Cristiano Hasenclever Borges  
Doutor em Engenharia Civil DCC - UFJF

Stênio São Rosário Furtado Soares  
Doutor em Ciência da Computação DCC - UFJF

JUIZ DE FORA  
11 DE JANEIRO, 2023

## Resumo

A demanda por serviços de entrega de mercadorias tem crescido muito nos últimos anos, principalmente por conta da popularização do comércio eletrônico. Diante desse novo cenário, o uso de drones em serviços de logística vem ganhando força. Os drones permitem que as entregas sejam feitas de forma mais rápida, uma vez que não são influenciados pelo trânsito nas grandes cidades e não dependem de infraestrutura rodoviária para a entrega (podem voar sobre áreas desabitadas). Além disso, como são alimentados, em sua maioria, por baterias elétricas, são mais econômicos e não emitem gases poluentes. Dessa forma, a introdução de drones em serviços de logística pode trazer benefícios para todos os envolvidos no processo. Neste trabalho será analisado o uso de drones em serviços de entregas de encomendas, implementando uma Busca Local Iterada (ILS) para tentar melhorar os resultados obtidos no trabalho de referência (DOAN; TAIYEB, 2020). A ideia é combinar drones e caminhões, a fim de otimizar o custo operacional.

**Palavras-chave:** Metaheurística, roteamento de drones, Busca Local Iterada, Colônia de Formigas.

## Abstract

The demand for delivery services has grown a lot in recent years, mainly due to the popularization of e-commerce. Given this new scenario, the use of drones in logistics services has been gaining strength. Drones allow deliveries to be made more quickly, since they are not influenced by traffic in large cities and do not depend on road infrastructure for delivery (they can fly over uninhabited areas). In addition, as they are mostly powered by electric batteries, they are more economical and do not emit polluting gases. In this way, the introduction of drones in logistics services can bring benefits to everyone involved in the process. In this work, the use of drones in parcel delivery services will be analyzed, implementing an Iterated Local Search (ILS) to try to improve the results obtained in the reference work (DOAN; TAIYEB, 2020). The idea is to combine drones and trucks in order to optimize operating costs.

**Keywords:** Metaheuristic, drone routing, Iterated Local Search, Ant Colony.

## Agradecimentos

À minha família e namorada pelo apoio em todos os momentos.

Aos professores Luciana, Stênio e Lorenza, pela atenção e carinho que me ofereceram.

Aos demais professores e funcionários da Universidade, por toda a ajuda e pelo conhecimento compartilhado.

Aos colegas de graduação, em especial, Gabriel Rezende e Pedro Eveling, pelo companheirismo durante esta jornada.

*“Você é o único representante do seu sonho na face da Terra”.*

*Emicida (Levanta e Anda)*

# Conteúdo

<b>Lista de Figuras</b>	<b>6</b>
<b>Lista de Tabelas</b>	<b>7</b>
<b>Lista de Abreviações</b>	<b>8</b>
<b>1 Introdução</b>	<b>9</b>
<b>2 Descrição do problema</b>	<b>14</b>
2.1 Notações utilizadas . . . . .	15
2.2 Descrição formal do problema . . . . .	17
2.3 Restrições do problema . . . . .	17
2.4 Parâmetros . . . . .	18
<b>3 Abordagens para o problema</b>	<b>20</b>
3.1 Clusterização dos Clientes . . . . .	20
3.2 Busca Local Iterada . . . . .	23
3.2.1 Movimento de busca local . . . . .	24
3.2.2 Movimentos de perturbação . . . . .	24
3.2.3 Critério de aceitação . . . . .	25
3.2.4 Critério de parada . . . . .	25
3.3 Abordagem da literatura . . . . .	26
<b>4 Experimentos Computacionais</b>	<b>28</b>
4.1 Instâncias . . . . .	28
4.2 Resultados . . . . .	29
4.3 Análise dos resultados . . . . .	37
<b>5 Conclusões</b>	<b>38</b>
<b>Bibliografia</b>	<b>40</b>

## Lista de Figuras

1.1	Drone do IFood operando em um <i>droneport</i> . . . . .	11
1.2	Drone aplicado em serviços de emergência médica . . . . .	12
2.1	Modelo do problema proposto . . . . .	15
2.2	Função objetivo do LRPAM . . . . .	17
3.1	Exemplo de cluster e atendimento aos clientes . . . . .	21
3.2	Algoritmo ILS . . . . .	24
3.3	<i>Swap-move</i> . . . . .	25
3.4	<i>Interchange-move</i> . . . . .	25

## Lista de Tabelas

2.1	Tabela de conjuntos . . . . .	15
2.2	Tabela de índices . . . . .	15
2.3	Tabela de variáveis de decisão . . . . .	16
2.4	Tabela de parâmetros utilizados . . . . .	16
2.5	Tabela de parâmetros do problema . . . . .	18
4.1	Tabela de instâncias utilizadas . . . . .	29
4.2	Melhores resultados para os cenários de 4Km x 4Km de área . . . . .	31
4.3	Melhores resultados para os cenários de 7,68Km x 7,68Km de área . . . . .	32
4.4	Melhores resultados para os cenários de 11Km x 11Km de área . . . . .	33
4.5	Resultados médios para os cenários de 4Km x 4Km de área . . . . .	34
4.6	Resultados médios para os cenários de 7,68Km x 7,68Km de área . . . . .	35
4.7	Resultados médios para os cenários de 11Km x 11Km de área . . . . .	36

## Lista de Abreviações

DCC	Departamento de Ciência da Computação
UFJF	Universidade Federal de Juiz de Fora
ICE	Instituto de Ciências Exatas
VANT	Veículo Aéreo Não Tripulado
ANAC	Agência Nacional de Aviação Civil
LRPAM	<i>Location Routing Problem with Ancillary Modes</i>
LRP	<i>Location Routing Problem</i>
ILS	Busca Local Iterada ( <i>Iterated Local Search</i> )
ACO	Algoritmo de Colônia de Formigas ( <i>Ant Colony Optimization</i> )
MACO	Algoritmo de Múltiplas Colônias de Formigas ( <i>Multiple Ant Colony Optimization</i> )
PRV	Problema de Roteamento de Veículos

# 1 Introdução

Nos últimos anos, a demanda pelos serviços de entrega de mercadorias aumentou de forma significativa, principalmente por conta da popularização do comércio eletrônico (GROUP, 2020). Em 2021, a Amazon divulgou que o serviço Prime, que oferece entrega gratuita em até 48 horas, já contava com mais de 150 milhões de assinantes em 19 países (ETUMNU, 2022). Diante desse novo cenário, o setor de logística passou a buscar soluções para atender, de forma mais eficiente, as demandas dos clientes.

Os desafios desse novo cenário incluem, entre outros, entregas com um tempo específico para serem realizadas, ou ainda a necessidade de adequar os serviços de logística às novas regulamentações ambientais. Estudos apontam que o setor de transportes é responsável por cerca de 24% das emissões globais de dióxido de carbono (CO<sub>2</sub>) (SO-LAYMANI, 2019). Nesse contexto, as discussões sobre o uso de drones (Veículos Aéreos Não-Tripulados - VANT) ganharam força. Isso porque esses dispositivos não emitem gases poluentes, uma vez que são movidos, em sua maioria, à baterias elétricas. Além disso, os drones permitem que pacotes sejam entregues com uma velocidade maior, visto que não dependem de uma infraestrutura física, como estradas e pontes, para se deslocarem, não são impactados por engarrafamentos no trânsito e são capazes de se deslocar em velocidades maiores. Outros pontos positivos em relação ao uso de drones são o reduzido custo operacional e o baixo consumo de energia. Estes dispositivos apresentam um reduzido custo operacional, sendo, normalmente, alimentados por baterias elétricas, não emitindo poluentes em virtude da queima de combustíveis fósseis, como outros veículos que compõem as redes logísticas. É possível combinar o uso de drones com diferentes modalidades de transporte, atendendo à demandas específicas com eficiência e otimização de custos. Portanto, os drones podem ser úteis para vencer os desafios do atual cenário dos serviços de entrega no mundo.

Porém, existe uma série de restrições e dificuldades na adoção do uso de drones em larga escala pelos serviços de logística. Dentre essas, vale destacar:

- Drones são muito afetados pelas condições climáticas. Portanto, os serviços seriam comprometidos em momentos de chuva, neve ou fortes rajadas de vento;
- Drones conseguem carregar uma pequena quantidade de carga e atender a poucos clientes em uma única viagem;
- O alcance dos drones é extremamente limitado, quando comparado ao alcance de outros modais;
- Nos grandes centros urbanos, são muitos os obstáculos ao deslocamento dos drones, como grandes construções, outdoors, viadutos e monumentos;
- Pode ser complexo encontrar uma maneira para o drone descarregar a encomenda. Normalmente, isso só é viável quando o cliente se encontra em um local aberto, como uma casa com jardim ou espaço público. A entrega em edifícios nos grandes centros urbanos é dificultada por essa situação;
- Drones ainda requerem, em sua maioria, que uma pessoa receba as encomendas, enquanto que, para entregas convencionais, o operador da entrega pode depositar a encomenda em um local apropriado, como uma caixa de correios (Sundar Kannan; Min, 2021).

Enquanto a questão da influência dos fatores climáticos na operação e a capacidade de carga reduzida são fatores inerentes aos drones, os pontos sobre obstáculos no ambiente urbano e dificuldades na entrega são fatores que podem ser solucionados, ou amenizados. Em relação ao deslocamento de drones no ambiente urbano, é possível a utilização de *softwares* de controle que tenham suas rotas automatizadas, de forma que os drones possam atingir seus destinos. Quanto à entrega das encomendas, pode-se recorrer ao uso de *lockers* para receber os produtos, ou o *delivery* ocorrendo em pontos específicos para o pouso de drones (*droneports*).

No Brasil, a primeira empresa autorizada a utilizar drones em serviços de entrega foi o IFood, em janeiro de 2022. De acordo com a regulamentação, os drones realizam uma etapa intermediária no processo de entrega, transportando a encomenda apenas entre

locais pré-estabelecidos e com uma infraestrutura para pousos e decolagens, denominados *droneports*. Entregadores humanos são responsáveis por coletar as encomendas nos *droneports* e levar até o cliente final. Todas as rotas são automatizadas e regulamentadas pela ANAC (Agência Nacional de Aviação Civil) (ANAC, 2022). A Figura 1.1 mostra um drone do IFood em operação.



Fonte: (GRUDIARIO, 2022)

Figura 1.1: Drone do IFood operando em um *droneport*

Considerando os desafios apontados, modelos que combinam o uso de drones com veículos convencionais, como caminhões, por exemplo, ganham força atualmente. Trabalhos como (HUANG et al., 2022; Sundar Kannan; Min, 2021; DOAN; TAIYEB, 2020; TAMKE; BUSCHER, 2022; POIKONEN; GOLDEN, 2020) propõem diferentes abordagens para combinar o uso de drones e caminhões. Enquanto alguns permitem entregas tanto por drones quanto por caminhões, outros definem que a última etapa das entregas deve ser realizada sempre por drones, com os caminhões sendo uma extensão do alcance dos drones.

Vale destacar que o uso de drones não é uma particularidade do setor de logística de transporte. Inicialmente desenvolvidos para uso militar, os drones podem, e já são, utilizados em diversos setores da sociedade. No setor de serviços de emergência, por exemplo, os drones são muito úteis em missões de busca e salvamento, principalmente após acidentes ou grandes catástrofes (KARACA et al., 2017), ou ainda na entrega de recursos médicos, como transporte de medicamentos, equipamentos, sangue e órgãos para transplantes (THIELS et al., 2015). Além disso, drones também são aplicados em atividades de irrigação e pulverização de produtos químicos em plantações, em atividades de vigilância,

como o monitoramento de áreas florestais e propriedades privadas. A Figura 1.2 mostra um exemplo de drone sendo utilizado em um serviço de emergência médica.



Fonte: <https://www.healthcareitnews.com/>

Figura 1.2: Drone aplicado em serviços de emergência médica

Os problemas de roteamento de veículos são um dos pilares dos sistemas de logística de entregas, existindo diferentes abordagens na literatura. Recentemente, o número de trabalhos que utilizam veículos auxiliares, como motos, trailers e drones, junto com os veículos primários (grandes caminhões), têm crescido muito, como em (SALAMA; SRINIVAS, 2022) e (WU et al., 2022).

Diante do exposto, o objetivo deste trabalho é abordar o uso de drones em conjunto com caminhões para realizar serviços de entregas para clientes. A ideia é expandir e melhorar os resultados encontrados no trabalho (DOAN; TAIYEB, 2020). No referido trabalho, os autores propuseram o uso de drones em conjunto com caminhões para realizar entregas para clientes e compararam os resultados com os mesmos cenários considerando apenas o uso de caminhões, visando a otimização dos custos operacionais. Neste trabalho, o problema abordado será o mesmo. Será proposta e implementada uma metaheurística para solucionar o problema, além de uma metaheurística nos mesmos moldes do trabalho base, para efeitos de comparação, em busca de melhores resultados.

O restante deste trabalho está dividido da seguinte maneira: O Capítulo 2 traz a descrição do problema; Em seguida, no Capítulo 3, são apresentadas as abordagens desenvolvidas; No Capítulo 4 são detalhados as instâncias utilizadas, os experimentos computacionais realizados, bem como os resultados obtidos; As conclusões e considerações

---

para trabalhos futuros são apresentadas no Capítulo 5.

## 2 Descrição do problema

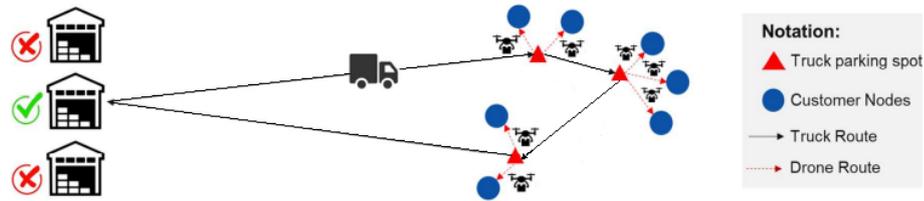
O problema abordado foi proposto em (DOAN; TAIYEB, 2020), tendo sido nomeado *Location Routing Problem with Ancillary Modes* (LRPAM). Na ocasião, os autores do trabalho base compararam os resultados de modelos com drones em conjunto com caminhões, e modelos apenas com caminhões (*Location Routing Problem - LRP*), em uma tentativa de evidenciar o impacto da introdução dos drones nos serviços de entrega.

Em relação ao LRPAM temos que, dado um conjunto de clientes com encomendas para receber, e um conjunto de caminhões que transportam encomendas e drones, o problema abordado trata da entrega dessas encomendas para os clientes finais, sendo que os drones realizam a última etapa das entregas.

As coordenadas dos clientes são conhecidas, e os caminhões devem partir de um dos depósitos disponíveis. Os caminhões estacionam em locais determinados, da onde lançam os drones que realizam a última etapa das entregas. Enquanto os caminhões ficam estacionados, os drones são lançados para atender os clientes próximos daquele estacionamento. Após realizar uma entrega, os drones retornam para o caminhão de origem, onde têm sua bateria trocada e são carregados com uma nova encomenda. Esse processo é repetido até que todos os clientes da vizinhança daquele estacionamento sejam atendidos. Após esse processo, o caminhão aguarda todos os drones retornarem para poder prosseguir. O caminhão pode retornar para o depósito de origem, bem como continuar para atender um próximo conjunto de clientes.

Vale ressaltar que a localização dos estacionamentos de lançamento de drones deve ser tal que os drones sejam capazes de percorrer a distância de ida e volta entre o caminhão e o cliente. Os cálculos e estratégias utilizadas para essas definições serão abordados no Capítulo 3. A Figura 2.1 ilustra o esquema de entregas para o problema:

Por ser um problema da classe NP-Difícil (DOAN; TAIYEB, 2020), o custo computacional para resolver grandes instâncias utilizando métodos exatos é elevado. Sendo assim, no Capítulo 3, são detalhadas duas metaheurísticas que tratam o problema, buscando uma boa solução em um tempo aceitável de computação.



Fonte: Adaptado de (DOAN; TAIYEB, 2020)

Figura 2.1: Modelo do problema proposto

Para o LRP, que seria o problema sem drones, temos que os caminhões saem dos depósitos selecionados e entregam as encomendas diretamente nas coordenadas dos clientes, não sendo feito nenhum tipo de cálculo de clusters. Neste trabalho, apenas o LRPAM foi modelado e estudado.

## 2.1 Notações utilizadas

As Tabelas 2.1, 2.2, 2.3 e 2.4, indicam, respectivamente, os conjuntos, os índices, as variáveis de decisão e os parâmetros que serão utilizado para a formulação matemática do LRPAM.

Tabela 2.1: Tabela de conjuntos

Símbolo	Descrição
D	conjunto dos $m$ depósitos
J	conjunto dos $n$ clientes
K	conjunto dos $k$ estacionamentos de lançamento

Tabela 2.2: Tabela de índices

Símbolo	Descrição
d	Representa um depósito
i, j	Representam clientes, depósitos e estacionamentos de lançamento
p	Representa estacionamentos de lançamento

Tabela 2.3: Tabela de variáveis de decisão

Símbolo	Descrição
$X_{ijd}$	Binário, vale 1 caso a aresta (i, d) seja percorrida partindo do depósito d
$Y_{ij}$	Variável contínua não negativa, indica a carga total restante no veículo antes de atingir o nó j na rota (i,j)

Tabela 2.4: Tabela de parâmetros utilizados

Símbolo	Descrição
$W_i$	Capacidade de um depósito $i$
$f_i$	Custo para abrir um depósito $i$
$Q_{max}$	Capacidade dos caminhões
$F_{drone}$	Custo fixo de uso de um drone por dia
$h$	Número de drones por caminhão
$F_{truck}$	Custo fixo de uso de um caminhão por rota
$C_i$	Custo de deslocamento de drone
$C_t$	Custo de deslocamento de caminhão
$C_{t-w}$	Custo de espera de um caminhão
$V_d$	Velocidade constante dos drones
$V_t$	Velocidade constante dos caminhões
$Dr$	Autonomia dos drones
$S_i$	Tempo de espera de um caminhão em um cluster
$T_l$	Tempo de carga de um drone
$T_u$	Tempo de descarga de um drone
$d_{ij}$	Distância entre os pontos $i$ e $j$

## 2.2 Descrição formal do problema

A Figura 2.2 indica a formulação matemática para a função objetivo do LRPAM.

$$\begin{aligned}
 Z = & \sum_{i=0}^n f_i y_i + F_{truck} \sum_{i=k+1}^{k+m} \sum_{j=1}^k \sum_{d=k+1}^{k+m} X_{ijd} \\
 & + \sum_{i=1}^{k+m} \sum_{j=1}^{k+m} \sum_{d=k+1}^{k+m} c_t \times d_{ij} \times X_{ijd} + \sum_{i=1}^k s'_i \times c_{t-w} \\
 & + \sum_{i=k+1}^{k+m} \sum_{j=1}^k X_{iji} \times F_{drone} \times h + \sum_{i=1}^p c'_i
 \end{aligned}$$

Fonte: Adaptado de (DOAN; TAIYEB, 2020)

Figura 2.2: Função objetivo do LRPAM

## 2.3 Restrições do problema

Foram feitas as seguintes definições para a modelagem do problema:

- Depósitos, estacionamentos e clientes estão contidos em seções delimitadas do Plano Cartesiano;
- Tanto a frota de caminhões, quanto a frota de drones, são homogêneas;
- A quantidade de carga, seja nos caminhões, seja nos drones, não influencia nos custos operacionais e nas velocidades;
- Todas as cargas são homogêneas;
- A demanda de todos os clientes é homogênea, sendo igual à uma carga;
- Cada drone é capaz de carregar apenas uma carga por vez;
- Apenas drones podem realizar a última etapa das entregas. Clientes não podem ser atendidos diretamente por caminhões;

- Drones apenas podem ser lançados de caminhões nos estacionamentos, não podendo ser lançados diretamente dos depósitos;
- O caminhão deve estar estacionado para lançar os drones, e deve permanecer assim até que todos os drones retornem;
- Os drones devem sempre retornar para seu caminhão de origem;
- Os caminhões devem sempre retornar para seu depósito de origem;
- As velocidades dos drones e dos caminhões são constantes.

## 2.4 Parâmetros

Os valores de algumas constantes são necessários para se trabalhar com o problema apresentado. A Tabela 2.5 indica os valores utilizados. Esses valores foram baseados no trabalho de referência (DOAN; TAIYEB, 2020).

Tabela 2.5: Tabela de parâmetros do problema

Símbolo	Descrição	Valor	Unidade de medida
$F_{drone}$	Custo fixo de uso de um drone	3.0769230	\$/drone/caminhão/dia
$F_{truck}$	Custo fixo de uso de um caminhão por rota	20.855944	\$/caminhão/rota/dia
$C_d$	Custo de deslocamento de drone	0.0000311	\$/metro
$C_t$	Custo de deslocamento de caminhão	0.0027968	\$/metro
$C_{t-w}$	Custo de espera de um caminhão	0.0059722	\$/segundo
$V_d$	Velocidade constante dos drones	16	metros/segundo
$V_t$	Velocidade constante dos caminhões	7	metros/segundo
$T_l$	Tempo de carga de um drone	20	segundos
$T_u$	Tempo de descarga de um drone	15	segundos
$h$	Número de drones por caminhão	3	
$K_t$	Fator de deslocamento do caminhão	2.46	-
$K_d$	Fator de deslocamento do drone	1	-

Os símbolos  $K_t$  e  $K_d$  indicam, respectivamente, os fatores de deslocamento dos caminhões e dos drones. Essas constantes são utilizadas como uma forma de tentar simular a liberdade que um drone tem para se deslocar no espaço aéreo, enquanto que os caminhões estão limitados à infraestrutura rodoviária existente. Dessa forma, ao calcular a distância entre dois pontos, o valor é multiplicado por  $K_t$ , ou por  $K_d$ , dependendo se está sendo analisado o deslocamento de um caminhão ou de um drone. Como o valor de  $K_d$  é

---

1, implica dizer que os drones percorrem trajetos em linha reta, entre um estacionamento de lançamento e um cliente.

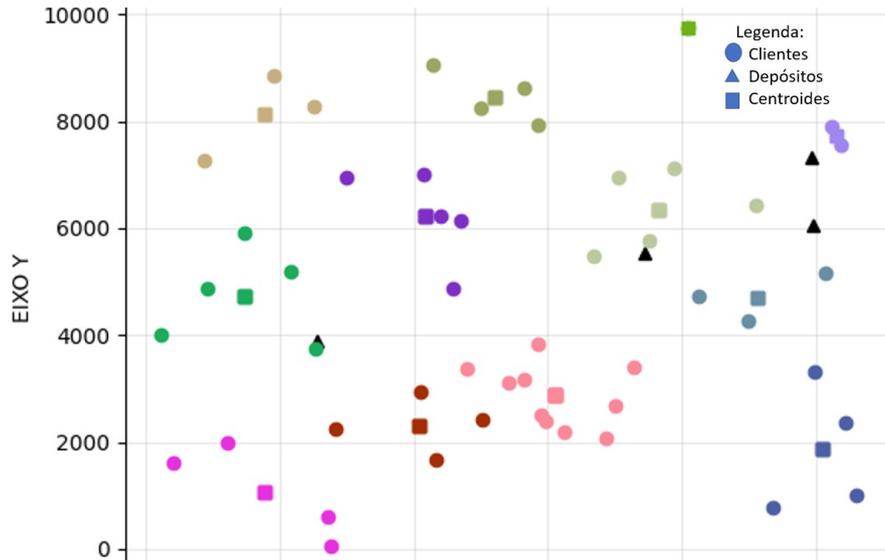
## 3 Abordagens para o problema

Para solucionar o problema foi desenvolvido um algoritmo baseado na metaheurística Busca Local Iterada (ILS - *Iterated Local Search*) proposta por (LOURENÇO; MARTIN; STÜTZLE, 2019). Para efeitos de comparação, foi desenvolvido um algoritmo Colônia de Formigas, nos mesmos moldes do trabalho de referência (DOAN; TAIYEB, 2020). Os dois algoritmos implementados partem de um ponto comum: o processo de clusterização dos clientes, descrito na próxima seção.

### 3.1 Clusterização dos Clientes

No processo de clusterização, os clientes, representados por pontos contidos em uma região limitada do Plano Cartesiano (Eixos X e Y), são agrupados (clusterizados) levando em conta a capacidade de carga do caminhão e a autonomia dos drones. Os drones, responsáveis pela última etapa das entregas, são lançados a partir de um estacionamento, que tem sua localização calculada durante o processo de clusterização. Os estacionamentos são localizados no centroide geográfico dos *clusters*. Logo, a autonomia dos drones deve ser suficiente para percorrer a distância de ida e volta entre o estacionamento e o cliente. Além disso, foi definido que um *cluster* não pode ser atendido de forma parcial no processo de entrega. Portanto, a demanda de um *cluster* não pode ser maior do que a capacidade de carga de um caminhão. O objetivo desta etapa é minimizar a quantidade de *clusters* gerados. O *dbscan* (KHAN et al., 2014) é um exemplo de outro algoritmo de clusterização que também visa reduzir a quantidade de *clusters* gerados. A Figura 3.1 mostra um exemplo de clusterização para uma instância do problema.

Uma vez executado o processo de clusterização, o problema restante se resume em rotear caminhões saindo dos depósitos e visitando todos os estacionamentos definidos, de forma que todos os clientes sejam atendidos. A partir deste ponto, os algoritmos implementados trabalham de forma independente. Nas próximas seções estão descritos de forma detalhada o funcionamento de cada um.



Fonte: Elaborado pelo autor (2022)

Figura 3.1: Exemplo de cluster e atendimento aos clientes

O algoritmo de clusterização desenvolvido utiliza como entrada uma lista com todos os pares de clientes presentes na instância, ordenada em ordem crescente de distância entre os clientes. Ao selecionar um par, é feita uma análise para descobrir se os clientes envolvidos já estão inseridos em algum *cluster*. Caso apenas um dos clientes esteja previamente inserido em um *cluster*, é feita uma tentativa de inserir o outro cliente nesse mesmo *cluster*, observando o número máximo de clientes em um *cluster* e se a autonomia do drone continua capaz de atender todos os clientes do novo *cluster*, visto que a localização do centróide vai ser alterada. Caso seja possível, essa inserção é efetuada. Caso não seja, o cliente que não estava inserido em nenhum *cluster* passa a constituir um *cluster* individual. Caso os dois clientes já estejam em *clusters* diferentes, é feita uma tentativa de união entre esses dois *clusters*, novamente observando a autonomia dos drones e o número máximo de clientes em um *cluster*. Caso seja possível, essa união é efetivada. Caso não seja, os clientes permanecem em seus *clusters* inalterados. Caso nenhum dos clientes esteja em um *cluster*, é criado, caso possível, um *cluster* com os dois clientes. Caso não seja possível, cada cliente passa a constituir um *cluster* individual. Esse processo é realizado até que todos os clientes estejam inseridos em *clusters*. A seguir temos um pseudocódigo que exemplifica o algoritmo de clusterização implementado:

**Algorithm** *Clusterizacao*

1  $C \leftarrow$  Lista de distância entre todos os pares de clientes, ordenada em ordem

```
    crescente
2  for each ( $a, b$ ) in  $C$  do
3      if  $a$  pertence a algum cluster e  $b$  não pertence then
4          Insere  $b$  no cluster de  $a$ 
5          if Cluster não viável then
6              Remove  $b$  do cluster
7               $b$  passa a constituir um novo cluster
8      else
9          if  $b$  pertence a algum cluster e  $a$  não pertence then
10             Insere  $a$  no cluster de  $b$ 
11             if Cluster não viável then
12                 Remove  $a$  do cluster
13                  $a$  passa a constituir um novo cluster
14             else
15                 if  $a$  e  $b$  pertencem a clusters diferentes then
16                     Une os dois clusters
17                     if Cluster resultado não viável then
18                         Desfaz a união entre os clusters
19                 else
20                     if  $a$  e  $b$  não pertencem a nenhum cluster then
21                         Cria novo cluster com  $a$  e  $b$ 
22                         if Cluster não viável then
23                             Deleta os clusters criados
24                              $a$  passa a constituir um novo cluster
25                              $b$  passa a constituir um novo cluster
26                         else
27                             if  $a$  pertence a algum cluster e  $b$  não pertence then
28                                 Insere  $b$  no cluster de  $a$ 
29                                 if Cluster não viável then
30                                      $b$  passa a constituir um novo cluster
```

## 3.2 Busca Local Iterada

O problema abordado pode ser dividido em dois subproblemas: o problema de localização, que envolve a clusterização dos clientes e associação entre *clusters* aos depósitos, e o problema de roteamento, que envolve a definição das rotas a serem realizadas por cada caminhão no processo de atendimento aos clientes dos *clusters*. Para solucionar essas duas etapas, foram implementados um algoritmo construtivo e um algoritmo de Busca Local Iterada.

O algoritmo construtivo implementado visa solucionar a primeira etapa do problema, associando *clusters* a depósitos abertos, e construindo uma solução inicial para que o algoritmo ILS possa refinar. Primeiramente, é necessário definir quais depósitos serão abertos. Para isso, para cada depósito, é calculada a soma das distâncias entre este e todos os estacionamentos de lançamento, de forma que o menor valor obtido indique o próximo depósito a ser aberto. Uma vez selecionado o depósito, os *clusters* mais próximos são associados a esse depósito e, caso a capacidade do depósito não permita que o *cluster* seja associado, é analisado o próximo *cluster* mais próximo, até que a capacidade do depósito seja utilizada em sua totalidade, ou até que a capacidade restante não seja suficiente para atender os *clusters* ainda não atendidos. Então, o depósito selecionado e os *cluster* associados são removidos das listas de candidatos disponíveis, e o processo é reiniciado, até que todos os *clusters* tenham sido associados a depósitos.

Ao final do algoritmo construtivo, a solução envolve um conjunto de depósitos a serem abertos e, cada um deles possui uma lista de *clusters* associados, além de rotas de entrega já definidas para cada depósito. Nessas rotas, os *clusters* são visitados na mesma ordem em que foram associados aos depósitos. Vale lembrar que os caminhões partem dos depósitos e estacionam nos centroides dos *clusters* para lançarem os drones, e aguardam no estacionamento de lançamento até todos os drones retornarem. Após atender um *cluster*, o caminhão pode prosseguir para atender um próximo *cluster*, ou retornar para o depósito de origem, para recarregar as encomendas ou terminar o serviço. O ILS desenvolvido será utilizado para encontrar melhores soluções para a etapa de roteamento dos caminhões.

A Busca Local Iterada (ILS - *Iterated Local Search*) é uma metaheu-

rística iterativa que alterna etapas de perturbação e busca local. O algoritmo envolve quatro principais componentes: uma solução inicial, uma estrutura de busca local, um movimento de perturbação e uma função critério de aceitação.

A partir de uma solução inicial, é realizada uma busca local para refinamento, que visa encontrar a solução ótima local na vizinhança da solução de referência. Tendo essa solução ótima local, é aplicado um movimento de perturbação nessa solução e, no resultado, é novamente aplicada uma busca local, visando um novo ótimo local. Caso a nova solução encontrada satisfaça a condição de critério de aceitação, ela é utilizada como base para uma nova rodada de perturbações e busca local. Esse processo é repetido até que um número pré-estabelecido de iterações seja executado. A Figura 3.2 mostra um pseudocódigo que representa a estrutura de um algoritmo ILS. A solução inicial  $s_0$  utilizada vem do algoritmo construtivo.

Algorithm	Template of the iterated local search algorithm.
	$s_* = \text{local search}(s_0) ;$ <b>Repeat</b> $s' = \text{Perturb}(s_*, \text{search history}) ;$ $s'_* = \text{Local search}(s') ;$ $s_* = \text{Accept}(s_*, s'_*, \text{search memory}) ;$ <b>Until</b> Stopping criteria <b>Output:</b> Best solution found.

Fonte: (TALBI, 2009)

Figura 3.2: Algoritmo ILS

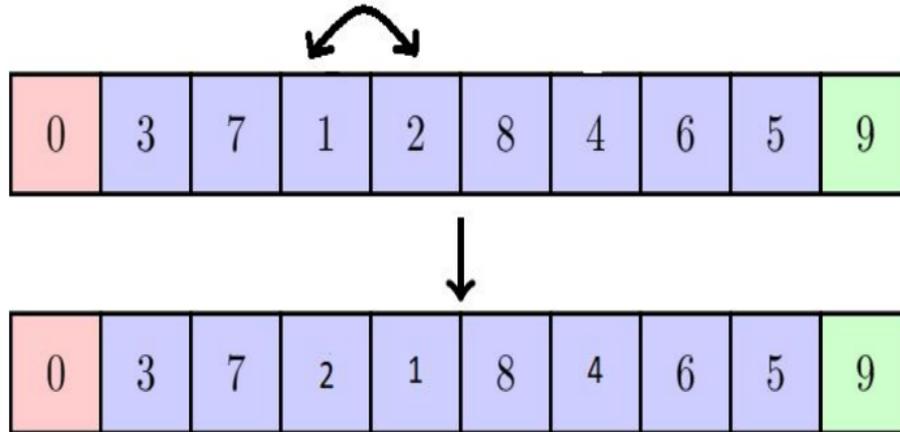
### 3.2.1 Movimento de busca local

O movimento da busca local utilizado neste trabalho foi a troca de posição entre dois *clusters* vizinhos na rota, explorando todas as soluções possíveis dentro da vizinhança estabelecida.

### 3.2.2 Movimentos de perturbação

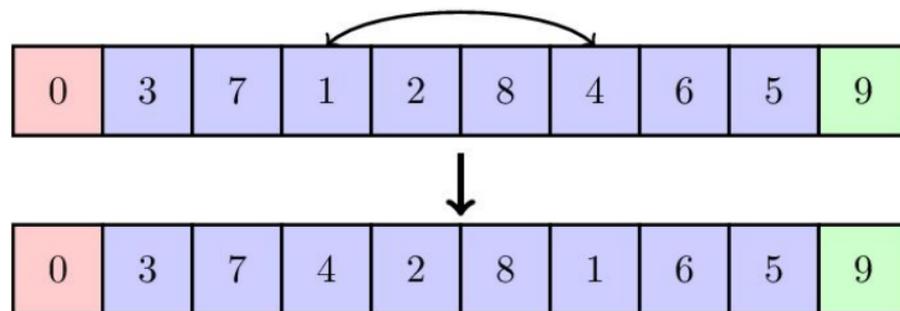
Como movimento de perturbação foi adotada uma combinação de duas trocas de posição entre elementos vizinhos aleatórios (*swap-moves*) e uma troca de posição entre elemento aleatórios (*interchange-moves*). Esses movimentos são detalhados em (SUBRAMANIAN,

2012). As Figuras 3.3 e 3.4 exemplificam, respectivamente, o *swap-move* e o *interchange-move*, aplicados em vetores de números inteiros.



Fonte: Elaborado pelo autor (2023)

Figura 3.3: *Swap-move*



Fonte: Elaborado pelo autor (2023)

Figura 3.4: *Interchange-move*

### 3.2.3 Critério de aceitação

O critério de aceitação utilizado para analisar os novos ótimos locais encontrados apenas aceita soluções com custo menor ou igual ao menor custo obtido até então.

### 3.2.4 Critério de parada

A partir do momento em que os *clusters* são associados aos depósitos, é possível observar cada associação como um problema independente. Portanto, o algoritmo ILS foi executado de forma separada para cada depósito. O critério de parada utilizado foi o número de iterações, que foi definido como 15.

### 3.3 Abordagem da literatura

Para efeito de comparação, foi implementado um algoritmo nos mesmos moldes do trabalho de referência, (DOAN; TAIYEB, 2020). Partindo de um cenário com os clientes já clusterizados, a abordagem da literatura enxerga o problema dividido em três etapas:

- Definir quantos, e quais, depósitos serão utilizados;
- Definir qual depósito atenderá cada *cluster*;
- Fazer o roteamento dos caminhões entre os depósitos e os estacionamentos de lançamentos de drones, localizados no centroide de cada *cluster*.

Cada uma dessas etapas é resolvida utilizando um algoritmo de Colônia de Formigas (ACO - *Ant Colony Optimization*) específico. Esse tipo de algoritmo utiliza técnicas probabilísticas inspiradas no comportamento real de formigas, que buscam o caminho mais curto até o alimento, através de trilhas de feromônio depositadas por outras formigas. Como serão utilizadas três colônias diferentes (uma para solucionar cada etapa do problema), o algoritmo aqui utilizado será denominado MACO (*Multiple Ant Colony Optimization*).

A primeira colônia de formigas foi desenvolvida para definir quantos, e quais, depósitos serão utilizados. Tendo o resultado desse primeiro algoritmo, a segunda colônia é responsável por associar os *clusters* aos depósitos, definindo de quais depósitos sairão os caminhões responsáveis por atender cada *cluster*. Após essas duas etapas, é necessário definir o roteamento de veículos, resolvendo um PRV, onde deve-se encontrar as rotas ideais, com base no menor custo, para os caminhões atenderem os *clusters*.

Nesta implementação do MACO, o algoritmo foi desenvolvido seguindo apenas uma breve descrição presente no trabalho de referência (DOAN; TAIYEB, 2020). Os autores do trabalho base não especificaram, em sua totalidade, as técnicas, as constantes e as variáveis utilizadas durante o desenvolvimento. Portanto, parâmetros fundamentais, como o número de formigas, número de iterações e o valor inicial de feromônio em cada matriz tiveram seus valores redefinidos neste trabalho. Foram utilizados, para

---

cada colônia, um valor inicial de feromônio igual a 1, um número de iterações igual a 10, e 10 formigas.

## 4 Experimentos Computacionais

Os códigos, tanto para o ILS, quanto para o MACO, foram implementados em C++20, enquanto que, no trabalho base, o desenvolvimento foi em Python 3.7. A execução dos códigos foi feita em um processador Intel I5-1135G7, 16Gb de memória RAM e Windows 10 Pro. Para cada teste, os algoritmos foram executados 10 vezes, tendo sido selecionadas as soluções que tenham gerado os menores custos e o tempo gasto. Foram coletados, ainda, o custo médio das soluções geradas e o tempo médio de execução.

### 4.1 Instâncias

Como os autores do trabalho de referência não disponibilizaram as instâncias utilizadas, foi desenvolvido um algoritmo gerador de instâncias. A ideia foi gerar instâncias nos mesmos moldes citados pelos autores. Portanto, o algoritmo gerou instâncias que compreendem pontos distribuídos em 3 diferentes tamanhos de seções de planos cartesianos: 4Km x 4Km, 7,68Km x 7,68Km e 11Km x 11Km. Para cada tamanho de área foram geradas 5 configurações de instância, cada uma com, respectivamente, 30, 40, 50, 200 e 300 clientes. A Tabela 4.1 traz os detalhes dos cenários gerados.

As colunas  $W_i$  e  $F_i$  indicam os intervalos nos quais são calculados, respectivamente, as capacidades e os custos de cada depósito da instância. A coluna  $Q_{max}$  indica a capacidade dos caminhões, em número de parcelas. Vale ressaltar que cada cliente possui demanda igual a uma parcela.

Para cada cenário foram geradas duas instâncias, totalizando 30 instâncias de teste. O código gerador de instâncias foi desenvolvido em C++20 e está disponível em [https://github.com/davirezende1998/tcc\\_engComp](https://github.com/davirezende1998/tcc_engComp).

Como a autonomia dos drones não foi informada no trabalho base, foram definidos dois valores para esse dado: 2Km e 3Km. Portanto, cada uma das 30 instâncias foi executada utilizando os dois valores de autonomia, resultando em 60 resultados gerados.

Tabela 4.1: Tabela de instâncias utilizadas

Cenário	Área	Clientes	Depósitos	fi	Wi	Qmax
1	4x4	30	3	[20, 40]	[100, 200]	10
2	4x4	40	3	[20, 40]	[100, 200]	10
3	4x4	50	4	[20, 40]	[100, 200]	10
4	4x4	200	5	[50, 100]	[300, 500]	25
5	4x4	300	5	[60, 100]	[300, 500]	25
6	7,68x7,68	30	3	[20, 40]	[400, 500]	10
7	7,68x7,68	40	3	[20, 40]	[400, 500]	10
8	7,68x7,68	50	4	[20, 40]	[400, 500]	10
9	7,68x7,68	200	5	[50, 100]	[600, 800]	25
10	7,68x7,68	300	5	[60, 100]	[600, 800]	25
11	11x11	30	3	[20, 40]	[600, 700]	10
12	11x11	40	3	[20, 40]	[600, 700]	10
13	11x11	50	4	[20, 40]	[600, 700]	10
14	11x11	200	5	[50, 100]	[800, 1000]	25
15	11x11	300	5	[50, 100]	[800, 1000]	25

## 4.2 Resultados

Nesta seção estão os resultados obtidos nos experimentos computacionais. As tabelas trazem os melhores resultados e as médias dos resultados obtidos para as instâncias, separadas conforme o tamanho da área. As tabelas possuem, para cada instância, duas linhas: uma com os resultados utilizando uma autonomia de drone igual à 2Km, e outra utilizando 3Km. As colunas indicam a quantidade de clientes, o alcance dos drones, o melhor resultado obtido para cada algoritmo e os respectivos tempos de execução.

Como as instâncias utilizadas foram criadas ao longo do projeto, não existem valores de resultados ótimos para consulta em literatura. Portanto, a coluna 'GAP' indica o quão melhor (ou pior) é o resultado obtido pelo ILS quando comparado ao resultado obtido pelo MACO desenvolvido nos moldes do trabalho base. A fórmula a seguir demonstra como foi feito o cálculo do GAP:  $GAP = (CustoILS - CustoMACO)/CustoMACO$

É possível observar na Tabela 4.2 que, para os 20 cenários onde a área é de tamanho 4Km x 4Km, o ILS obteve melhor resultado que o MACO em 15. Em três instâncias houve empate e em apenas duas o MACO foi melhor que a solução do ILS. Porém, enquanto o tempo de execução médio do MACO foi de 5,98 segundos, o ILS precisou, em média, de 6,76 segundos para encontrar a solução.

Na Tabela 4.3, para o cenário de área 7,68Km x 7,68Km, em 20% dos melhores resultados houve empate entre os melhores valores encontrados pelos dois algoritmos. Essa quantidade é maior do que a observada para as instâncias de dimensões 4Km x 4Km, quando empates ocorreram em apenas 15% dos casos. Em 12 dos 20 resultados apresentados nesta tabela, o ILS obteve resultado melhor que o obtido pelo MACO.

Para os experimentos com instâncias no cenário com dimensões 11Km x 11Km, registradas na Tabela 4.4, é possível observar que em 13 instâncias a solução do ILS é melhor que a solução do MACO, e que os algoritmos empataram em 5 instâncias. Sobre o tempo de execução, em apenas um momento (5% dos casos), o tempo do ILS foi menor do que o tempo de execução do MACO. Esse valor foi o mesmo observado para as instâncias de 7,68Km x 7,68Km, enquanto que, para as instâncias de 4Km x 4Km, o ILS foi mais rápido do que o MACO em 10% dos casos.

Analisando a média das 10 execuções para as instâncias de 4Km x 4Km, na Tabela 4.5, é possível observar que em apenas um caso o ILS não conseguiu obter um custo médio menor do que o custo médio do MACO. Sobre o tempo médio de processamento, o ILS consome 7,03 segundos, enquanto que o MACO leva 5,88 segundos para encontrar a solução final.

Analisando as Tabelas 4.6 e 4.7, tanto para o cenário de 7,68Km x 7,68Km, quanto para o cenário de 11Km x 11Km, observa-se que em 100% dos casos o ILS conseguiu encontrar custos médios menores do que o MACO. Quando analisados os tempos de execução, observa-se que em 10% das ocorrências o ILS foi mais rápido do que o MACO. Enquanto, para as instâncias de 7,68Km x 7,68Km, o tempo médio do ILS foi 10,20 segundos e o tempo médio do MACO foi 8,92 segundos, para as instâncias de 11Km x 11Km o tempo médio do ILS foi 9,46 segundos, e o tempo médio do MACO foi 7,97 segundos.

Tabela 4.2: Melhores resultados para os cenários de 4Km x 4Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	312,99	343,53	<b>-0,09</b>	998	589
inst1_30	30	3	405,40	476,35	<b>-0,15</b>	1090	783
inst2_30	30	2	501,72	582,04	<b>-0,14</b>	1004	829
inst2_30	30	3	400,77	429,76	<b>-0,07</b>	984	1017
inst1_40	40	2	510,04	510,04	0,00	1267	1006
inst1_40	40	3	401,90	455,93	<b>-0,12</b>	1463	1280
inst2_40	40	2	531,98	525,06	0,01	1332	1491
inst2_40	40	3	502,67	581,80	<b>-0,14</b>	1120	912
inst1_50	50	2	691,24	750,23	<b>-0,08</b>	2310	1832
inst1_50	50	3	643,09	650,27	<b>-0,01</b>	2543	2090
inst2_50	50	2	651,29	726,27	<b>-0,10</b>	2753	2173
inst2_50	50	3	590,14	560,32	0,05	2619	2459
inst1_200	200	2	2282,18	2282,18	0,00	10765	10297
inst1_200	200	3	1672,10	1926,39	<b>-0,13</b>	10196	8916
inst2_200	200	2	1811,22	2096,10	<b>-0,14</b>	12107	10939
inst2_200	200	3	1873,59	2872,66	<b>-0,35</b>	11573	10477
inst1_300	300	2	2390,71	2639,18	<b>-0,09</b>	17491	14993
inst1_300	300	3	2210,76	2519,33	<b>-0,12</b>	16501	14722
inst2_300	300	2	2437,74	2437,74	0,00	18932	16011
inst2_300	300	3	2177,81	2381,66	<b>-0,09</b>	18101	16734
<b>MÉDIA</b>	-	-	<b>1149,97</b>	<b>1287,34</b>	<b>-0,09</b>	<b>675,45</b>	<b>5977,50</b>

Tabela 4.3: Melhores resultados para os cenários de 7,68Km x 7,68Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	1349,56	1349,56	0,00	1934	1072
inst1_30	30	3	940,97	923,37	0,02	1988	1291
inst2_30	30	2	1239,02	1344,95	<b>-0,08</b>	1526	981
inst2_30	30	3	1401,91	1311,46	0,07	1671	1388
inst1_40	40	2	1379,12	1531,27	<b>-0,10</b>	2371	1995
inst1_40	40	3	1350,50	1471,22	<b>-0,08</b>	2410	2022
inst2_40	40	2	1541,60	1541,60	0,00	2871	2396
inst2_40	40	3	1371,01	1485,43	<b>-0,08</b>	2941	1927
inst1_50	50	2	1630,92	1775,31	<b>-0,08</b>	3298	3506
inst1_50	50	3	1578,68	1578,68	0,00	3619	2492
inst2_50	50	2	1432,98	1493,91	<b>-0,04</b>	3918	2917
inst2_50	50	3	1389,71	1367,19	0,02	3817	3202
inst1_200	200	2	3807,88	4062,91	<b>-0,06</b>	20729	16339
inst1_200	200	3	3109,86	3389,53	<b>-0,08</b>	20587	17091
inst2_200	200	2	3910,22	3348,53	0,17	19628	17288
inst2_200	200	3	3623,21	3623,21	0,00	19504	18071
inst1_300	300	2	3801,77	4103,87	<b>-0,07</b>	22691	20187
inst1_300	300	3	3691,80	3829,06	<b>-0,04</b>	23930	22551
inst2_300	300	2	4005,91	4295,82	<b>-0,07</b>	24301	21397
inst2_300	300	3	3871,33	4071,31	<b>-0,05</b>	23945	20410
<b>MÉDIA</b>	-	-	<b>2321,34</b>	<b>2394,91</b>	<b>-0,03</b>	<b>10383,95</b>	<b>8926,15</b>

Tabela 4.4: Melhores resultados para os cenários de 11Km x 11Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	1237,86	1237,86	0,00	1309	971
inst1_30	30	3	1009,11	1293,02	<b>-0,22</b>	1586	990
inst2_30	30	2	1739,01	1943,22	<b>-0,11</b>	1601	1246
inst2_30	30	3	1302,81	1224,64	0,06	1992	1820
inst1_40	40	2	2045,82	2045,82	0,00	2093	2250
inst1_40	40	3	1762,34	1962,07	<b>-0,10</b>	2108	1729
inst2_40	40	2	1790,51	1998,03	<b>-0,10</b>	3018	2510
inst2_40	40	3	1649,02	1961,62	<b>-0,16</b>	2971	2009
inst1_50	50	2	1981,21	2276,69	<b>-0,13</b>	3816	2899
inst1_50	50	3	1971,01	2147,72	<b>-0,08</b>	3566	3002
inst2_50	50	2	2001,71	2236,04	<b>-0,10</b>	4012	3171
inst2_50	50	3	2186,08	2186,08	0,00	3819	3598
inst1_200	200	2	4108,73	4419,29	<b>-0,07</b>	17376	14990
inst1_200	200	3	4050,66	4016,05	0,01	17941	15622
inst2_200	200	2	4386,09	4386,09	0,00	16302	13977
inst2_200	200	3	3813,90	4080,25	<b>-0,07</b>	16947	14002
inst1_300	300	2	4552,91	4956,72	<b>-0,08</b>	22618	18021
inst1_300	300	3	4329,79	4722,19	<b>-0,08</b>	21190	19773
inst2_300	300	2	5210,88	5382,10	<b>-0,03</b>	23346	20138
inst2_300	300	3	5102,89	5102,89	0,00	22931	21004
<b>MÉDIA</b>	-	-	<b>2811,62</b>	<b>2978,92</b>	<b>-0,06</b>	<b>9527,10</b>	<b>8186,10</b>

Tabela 4.5: Resultados médios para os cenários de 4Km x 4Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	501,85	555,19	-0,10	1016	996
inst1_30	30	3	498,01	587,84	-0,15	1054	761
inst2_30	30	2	708,93	722,24	-0,02	1043	1004
inst2_30	30	3	<b>605,66</b>	<b>533,60</b>	<b>0,14</b>	1044	871
inst1_40	40	2	653,09	714,32	-0,09	<b>1302</b>	<b>1401</b>
inst1_40	40	3	508,08	606,69	-0,16	1356	1210
inst2_40	40	2	592,30	645,54	-0,08	<b>1250</b>	<b>1305</b>
inst2_40	40	3	580,24	653,85	-0,11	1391	1242
inst1_50	50	2	751,04	867,71	-0,13	2485	2091
inst1_50	50	3	691,80	780,09	-0,11	2618	2271
inst2_50	50	2	734,01	890,91	-0,18	2701	1903
inst2_50	50	3	701,31	811,24	-0,14	2881	2090
inst1_200	200	2	2438,01	2685,81	-0,09	11308	10991
inst1_200	200	3	1782,10	2347,18	-0,24	10958	10706
inst2_200	200	2	2491,77	2519,40	-0,01	12884	9206
inst2_200	200	3	2561,03	3207,59	-0,20	12199	9014
inst1_300	300	2	2801,75	2913,87	-0,04	17934	15001
inst1_300	300	3	2408,77	2786,49	-0,14	17204	13839
inst2_300	300	2	2634,55	2806,97	-0,06	19103	15003
inst2_300	300	3	2610,72	2940,44	-0,11	18882	16602
<b>MÉDIA</b>	-	-	<b>1362,75</b>	<b>1528,85</b>	<b>-0,10</b>	<b>7030,65</b>	<b>5875,35</b>

Tabela 4.6: Resultados médios para os cenários de 7,68Km x 7,68Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	1509,98	1897,35	-0,20	2003	882
inst1_30	30	3	1023,91	1358,85	-0,25	1931	1073
inst2_30	30	2	1455,60	1538,45	-0,05	1769	1045
inst2_30	30	3	1501,29	1704,80	-0,12	1823	1421
inst1_40	40	2	1450,02	1962,10	-0,26	2506	2277
inst1_40	40	3	1590,19	1711,36	-0,07	2737	2008
inst2_40	40	2	1903,99	2079,90	-0,08	2591	1855
inst2_40	40	3	1688,32	1776,42	-0,05	2855	2309
inst1_50	50	2	1901,87	2028,55	-0,06	<b>3102</b>	<b>3105</b>
inst1_50	50	3	1704,39	2025,26	-0,16	3216	2122
inst2_50	50	2	1691,20	1823,77	-0,07	3711	2139
inst2_50	50	3	1822,30	2035,89	-0,10	3466	2590
inst1_200	200	2	4201,90	4766,57	-0,12	20103	17981
inst1_200	200	3	3766,51	4185,36	-0,10	21011	16059
inst2_200	200	2	<b>4203,90</b>	<b>3740,89</b>	<b>0,12</b>	19993	16502
inst2_200	200	3	4010,96	4290,18	-0,07	19951	18772
inst1_300	300	2	4203,10	4506,15	-0,07	22012	18970
inst1_300	300	3	3807,69	4012,71	-0,05	22939	21348
inst2_300	300	2	4209,81	4329,76	-0,03	23104	22991
inst2_300	300	3	4005,69	4321,90	-0,07	23106	22870
<b>MÉDIA</b>	-	-	<b>2582,63</b>	<b>2804,81</b>	<b>-0,09</b>	<b>10196,45</b>	<b>8915,95</b>

Tabela 4.7: Resultados médios para os cenários de 11Km x 11Km de área

Instância			Custo total			Tempo total (ms)	
ID	Clientes	Dr (Km)	ILS	MACO	Gap	ILS	MACO
inst1_30	30	2	1504,90	1795,44	-0,16	1498	1301
inst1_30	30	3	1807,66	2160,83	-0,16	1288	1255
inst2_30	30	2	2013,40	2549,26	-0,21	1377	1105
inst2_30	30	3	1506,01	1727,67	-0,13	<b>1481</b>	<b>2111</b>
inst1_40	40	2	2309,75	2498,13	-0,08	<b>2361</b>	<b>3004</b>
inst1_40	40	3	2096,06	2517,17	-0,17	1993	1971
inst2_40	40	2	2107,88	2510,12	-0,16	2849	2307
inst2_40	40	3	2504,66	2822,92	-0,11	2800	1833
inst1_50	50	2	2307,99	2869,06	-0,20	3249	2308
inst1_50	50	3	2503,23	2810,55	-0,11	3108	2761
inst2_50	50	2	2704,23	2984,44	-0,09	3739	2779
inst2_50	50	3	2403,88	2860,81	-0,16	3208	3720
inst1_200	200	2	5492,24	5703,68	-0,04	17106	13882
inst1_200	200	3	4533,31	4827,29	-0,06	17720	14839
inst2_200	200	2	5087,15	5470,08	-0,07	16965	15003
inst2_200	200	3	4511,79	4964,31	-0,09	17439	14301
inst1_300	300	2	5066,90	5379,55	-0,06	20959	17068
inst1_300	300	3	4800,38	5094,23	-0,06	22544	19881
inst2_300	300	2	5331,09	5621,46	-0,05	23971	18052
inst2_300	300	3	5299,74	5371,69	-0,01	23533	19911
<b>MÉDIA</b>	-	-	<b>3294,61</b>	<b>3626,93</b>	<b>-0,12</b>	<b>9459,40</b>	<b>7969,60</b>

## 4.3 Análise dos resultados

É possível observar que o algoritmo ILS encontrou, para a maioria das instâncias, resultados melhores (custos menores), quando comparado ao algoritmo MACO. Como os autores do trabalho base omitiram informações importantes sobre o algoritmo, como o número de formigas, número de iterações e quantidade inicial de feromônio em cada colônia, essa diferença nos resultados obtidos pode ser consequência dos valores definidos para estas constantes neste trabalho. A diferença nos valores encontrados fica nítida quando a coluna 'GAP' é analisada. É perceptível que a maior parte dos valores são 'gaps negativos', indicando que o resultado obtido pelo ILS foi melhor do que o resultado obtido pelo MACO.

Em relação aos tempos de execução, o algoritmo MACO possui, em sua maioria, uma execução mais rápida do que o algoritmo ILS. Isso se deve, em parte, à busca local do algoritmo, que demanda um maior tempo de processamento. Ainda nesse ponto, temos que, para cada solução parcial obtida, é feito um cálculo de função objetivo, além de diversos cálculos de viabilidade ao longo do processo de obtenção dessa solução parcial, visando respeitar todas as restrições do problema. Todos esses cálculos fazem com que o custo computacional e, conseqüentemente, o tempo de execução, de todos os algoritmos seja aumentado.

É possível perceber, ainda, que uma autonomia de drone (*drone range*) maior leva a soluções com um menor custo total, em sua maioria. Isso porque com uma autonomia maior, é possível reduzir o número de *clusters*. Dessa forma, a solução final envolve menos deslocamentos de caminhões entre *clusters*, e mais deslocamentos de drones dentro de *clusters*. Como o deslocamento de drones é significativamente mais barato que o deslocamento de caminhões, a tendência é que o custo final dessas soluções seja menor.

## 5 Conclusões

A demanda pelos serviços de entrega aumentou muito nos últimos anos, e tende a crescer cada vez mais. Diante deste crescimento, os novos desafios que surgem incluem aumentar a eficiência das entregas, reduzindo custos operacionais e adequando os serviços às novas regulamentações ambientais. Com base nessas novas necessidades, foi analisado o impacto do uso de drones, em conjunto aos caminhões, para a realização dos serviços de entrega. A ideia era reduzir os custos operacionais, aumentar a velocidade das entregas e reduzir as emissões de dióxido de carbono ( $\text{CO}_2$ ) na atmosfera. Para isso, foi desenvolvido um algoritmo Colônia de Formigas (MACO), nos mesmos moldes da literatura, e um algoritmo ILS, para tentar melhorar os resultados obtidos pelos autores do trabalho de referência. Vale ressaltar que os autores do trabalho base não forneceram as instâncias utilizadas, e omitiram dados importantes do algoritmo utilizado, como o número de formigas, o número de iterações, o nível inicial de feromônio e a autonomia dos drones. Com isso, foi necessário definir novos valores e constantes neste projeto, e propor novas instâncias para o problema.

Sobre os resultados obtidos, é possível observar que os custos obtidos pelo algoritmo ILS proposto foram, em sua maioria, menores do que os custos obtidos pelo algoritmo MACO. Em relação aos tempos de execução, o ILS possui, para a maior parte das instâncias, um tempo de execução maior do que o MACO. A estrutura de busca local do ILS consome a maior parte do tempo de execução desse algoritmo. É importante ressaltar que o tempo de execução coletado para o ILS foi o equivalente à todas as 15 iterações (critério de parada) e, como a tendência do ILS é se aproximar cada vez mais de uma solução ótima, não é possível garantir em qual iteração foi encontrada a solução final.

Uma característica marcante do projeto foi a estratégia utilizada para a clusterização dos clientes, feita segundo o trabalho base. Tanto o ILS, quanto o MACO, se valeram dessa estrutura. Um ponto particular dessa etapa é a localização dos estacionamentos de lançamento de drones, que têm suas coordenadas calculadas e alte-

radas em tempo de execução, com base nas coordenadas dos clientes que compõe aquele *cluster*. Em um cenário real, essa situação pode ser inviável, visto que em grandes centros urbanos pode não ser possível instalar um estacionamento para lançamento de drones em qualquer ponto que seja calculado. Portanto, para trabalhos futuros, pode ser interessante trabalhar com outras formas de clusterização e localização desses pontos. Vale ressaltar que, a clusterização utilizada neste trabalho pode ser custosa computacionalmente, uma vez que é possível que ocorram diversas uniões entre clusters, o que envolve muitas operações para verificação das restrições. Em relação à localização, uma ideia é utilizar uma lista de estacionamentos com coordenadas pré-definidas, e realizar o processo de clusterização em função dessas localidades.

Foi ressaltado ao longo do trabalho que dados importantes foram omitidos no trabalho base, fazendo com que novos valores fossem definidos e utilizados aqui. Foi o caso de nível inicial de feromônio para as colônias do MACO, bem como da quantidade de formigas e da quantidade de iterações, além da autonomia dos drones. Não foi feito um estudo aprofundado sobre os valores aqui definidos. Dessa forma, para trabalhos futuros, sugere-se que seja feita uma análise detalhada sobre o impacto dos valores utilizados, afim de se calibrar melhor essas constantes, na expectativa de que o algoritmo encontre resultados melhores. Com relação ao número de formigas, por exemplo, foram utilizadas 10 formigas em cada uma das três colônias do MACO, mas pode ser interessante que cada colônia tenha uma quantidade específica de formigas. Um estudo sobre estes valores pode melhorar o resultado do algoritmo.

Por fim, sobre o ILS desenvolvido, tem-se que um algoritmo construtivo soluciona as duas primeiras etapas do problema (seleção de depósitos e associação de *clusters* a depósitos), e não há nenhuma alteração posterior nessas decisões iniciais. Ou seja, uma vez que esse algoritmo construtivo é executado, as soluções utilizadas como base para a etapa do roteamento não se alteram. Em trabalhos futuros é importante que os movimentos de perturbação do ILS caminhem em direção a uma visita às definições das primeiras etapas, de forma que mudanças possam ser feitas, aumentando as possibilidades de visitar novas soluções, em busca de obter melhores resultados.

## Bibliografia

- ANAC. *ANAC concede a primeira autorização para entregas comerciais com drone*. 2022.
- DOAN, V. B. N.; TAIYEB, A. M. Facility and routing decisions in truck-drone distribution. In: . [S.l.]: MIT, 2020.
- ETUMNU, C. E. Free shipping. *Applied Economics Letters*, Routledge, v. 0, n. 0, p. 1–4, 2022. Disponível em: <<https://doi.org/10.1080/13504851.2022.2094871>>.
- GROUP, I. *Logistics Market: Global Industry Trends, Share, Size, Growth, Opportunity and Forecast 2023-2028*. 2020. Url<https://www.imargroup.com/logistics-market>.
- GRUDIARIO. *Anac autoriza iFood a fazer entregas com drones*. 2022. Disponível em: <<https://www.grudiario.com.br/anac-autoriza-ifood-a-fazer-entregas-com-drones>>.
- HUANG, S.-H. et al. Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm. *Advanced Engineering Informatics*, v. 51, p. 101536, 2022. ISSN 1474-0346. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S147403462200012X>>.
- KARACA, Y. et al. The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations. In: . [S.l.]: Elsevier, 2017. p. 585–588.
- KHAN, K. et al. Dbscan: Past, present and future. In: *The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014)*. [S.l.: s.n.], 2014. p. 232–238.
- LOURENÇO, H. R.; MARTIN, O. C.; STÜTZLE, T. Iterated local search: Framework and applications. In: \_\_\_\_\_. *Handbook of Metaheuristics*. Springer International Publishing, 2019. p. 129–168. ISBN 978-3-319-91086-4. Disponível em: <[https://doi.org/10.1007/978-3-319-91086-4\\_5](https://doi.org/10.1007/978-3-319-91086-4_5)>.
- POIKONEN, S.; GOLDEN, B. Multi-visit drone routing problem. *Computers Operations Research*, v. 113, p. 104802, 2020. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054819302448>>.
- SALAMA, M. R.; SRINIVAS, S. Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites. *Transportation Research Part E: Logistics and Transportation Review*, v. 164, p. 102788, 2022. ISSN 1366-5545. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1366554522001776>>.
- SOLAYMANI, S. Co2 emissions patterns in 7 top carbon emitter economies: The case of transport sector. *Energy*, v. 168, p. 989–1001, 2019. ISSN 0360-5442. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0360544218323569>>.
- SUBRAMANIAN, A. *Heuristic, Exact and Hybrid Approaches for Vehicle*. Tese (Doutorado) — Tesis Doctoral, Niterói, 2012.
- Sundar Kannan, S.; Min, B.-C. Autonomous Drone Delivery to Your Door and Yard. *arXiv e-prints*, p. arXiv:2104.05503, abr. 2021.

---

TALBI, E.-G. *Metaheuristics - From Design to Implementation*. [S.l.]: Wiley, 2009. I-XXIX, 1-593 p. ISBN 978-0-470-27858-1.

TAMKE, F.; BUSCHER, U. The vehicle routing problem with drones and drone speed selection. *Computers Operations Research*, p. 106112, 2022. ISSN 0305-0548. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0305054822003422>>.

THIELS, C. A. et al. Use of unmanned aerial vehicles for medical product transport. In: . [S.l.]: Elsevier, 2015. p. 104–108.

WU, G. et al. Collaborative truck-drone routing for contactless parcel delivery during the epidemic. *IEEE Transactions on Intelligent Transportation Systems*, v. 23, n. 12, p. 25077–25091, 2022.