

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**UM MÉTODO DE PLANEJAMENTO DE ROTAS  
DE VOO DE VANT MULTIROTOR PARA  
COBERTURA DE ÁREAS UTILIZANDO A  
META-HEURÍSTICA ACO**

**LUCAS DOS SANTOS FRANCO**

**ORIENTADOR: PROF. DR. EDILSON REIS RODRIGUES KATO**

São Carlos – SP

Dezembro/2019

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**UM MÉTODO DE PLANEJAMENTO DE ROTAS  
DE VOO DE VANT MULTIROTOR PARA  
COBERTURA DE ÁREAS UTILIZANDO A  
META-HEURÍSTICA ACO**

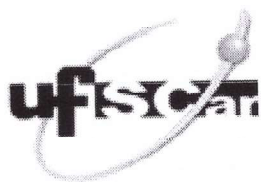
**LUCAS DOS SANTOS FRANCO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação, área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Edilson Reis Rodrigues Kato

São Carlos – SP

Dezembro/2019



# UNIVERSIDADE FEDERAL DE SÃO CARLOS


Centro de Ciências Exatas e de Tecnologia  
Programa de Pós-Graduação em Ciência da Computação

---

## Folha de Aprovação

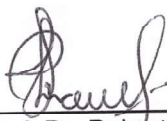
---

Assinaturas dos membros da comissão examinadora que avaliou e aprovou a Defesa de Dissertação de Mestrado do candidato Lucas dos Santos Franco, realizada em 09/12/2019:



---

Prof. Dr. Edilson Reis Rodrigues Kato  
UFSCar



---

Prof. Dr. Roberto Santos Inoue  
UFSCar



---

Prof. Dr. Mário Luiz Tronco  
USP

A minha companheira Cássia, aos meus pais e a minha irmã, que me incentivaram durante essa fase e me deram todo o suporte necessário nos momentos difíceis. Ao Marcelo Selli por ter me apresentado ao mundo da pesquisa e principalmente por ter compartilhado tantos ensinamentos de vida.

## AGRADECIMENTOS

Ao meu Orientador Edilson Kato, que desde o início entendeu e confiou na minha forma de trabalhar e estudar, e com base nisto me orientou da melhor maneira possível nesta etapa, sempre me acalmando e me mostrando alternativas quando eu chegava preocupado com algo.

Ao Fernando Guimarães Aguiar, o Fernandinho, que sempre me incentivou a ingressar na carreira acadêmica e que inclusive me acolheu em sua casa nas minhas primeiras semanas em São Carlos.

Ao Lúcio Jorge por me trazer mais para perto da área acadêmica e pela oportunidade que me deu de contribuir com suas pesquisas e trabalhar junto com a equipe do LANAPRE.

A toda equipe do LANAPRE, em especial ao time técnico, pelo convívio diário que resultou em uma coleção de momentos ímpares recheado de conhecimento e amizades.

Aos professores, servidores, e alunos do PPGCC que fazem do departamento um lugar muito acolhedor e propício para a disseminação de conhecimento.

Ao aluno Emmanuel Marrocos, que me acolheu em seu grupo de estudos, e passou de um parceiro de aula para um amigo pra vida.

Agradeço aos meus pais que me deram todo incentivo e suporte para iniciar e concluir este estudo. Sem esse apoio nada seria possível.

Agradeço a minha irmã por compartilhar comigo seu conhecimento e experiência do meio acadêmico, foi tudo muito útil.

Agradeço minha companheira, Cássia, por sempre me impulsionar a crescer como profissional e como pessoa, por ser minha conselheira nos momentos de incerteza, por não me deixar abater nas dificuldades e por entender a minha ausência nesse período de estudos.

*What's measured improves*

Peter Drucker

## RESUMO

Com a popularização dos Veículos Aéreos Não Tripulados (VANTs), e com a ampliação das áreas de aplicação dessa tecnologia, nota-se um aumento de pesquisas com foco no planejamento de rotas de voo. Este trabalho parte da utilização de VANTs em cenários agrícolas na tarefa de imageamento aéreo. O objetivo é apresentar um método de planejamento de rota de cobertura (*Coverage Path Planning* - CPP) para VANTs multirotor e comparar este método com uma solução já utilizada no mercado. O método desenvolvido considera cenários com múltiplos terrenos, e busca apresentar uma proposta de rota com otimização de ordem de visita dos terrenos, minimização do número de curvas e da distância total do percurso. O método apresentado contém três etapas principais. A primeira trata da decomposição das áreas, onde terrenos representados por polígonos côncavos são decompostos em subáreas menores de formato convexo através da utilização de um algoritmo guloso. A segunda etapa calcula a direção de voo que minimiza o número de curvas de trajeto em cada uma das subáreas, encontrando a direção da maior altura do polígono para orientar o sentido da aplicação de padrão de movimentação de ida e vinda, conhecido na literatura por padrão *boustrophedon*. A terceira etapa trata da otimização da ordem de visita das subáreas, nesta fase o cenário é modelado como uma especialização do Problema de Caixeiro Viajante, conhecida como Caixeiro Viajante Generalizado (*Generalized Travelling Salesman Problem* - GTSP), e para solucionar este problema é utilizado o algoritmo de Otimização por Colônia de Formigas (*Ant Colony Optimization* - ACO). Os resultados obtidos através do método proposto são comparados com as soluções propostas por um programa de planejamento de rota já utilizado pelo mercado, o *Mission Planner*. Para medir a eficiência das soluções foram consideradas duas variáveis: a distância total percorrida e o número de curvas da rota. Através dos resultados obtidos pôde-se mapear os tipos de cenários onde o método desenvolvido pode agregar com as soluções de mercado.

**Palavras-chave:** VANT, Agricultura de Precisão, Planejamento de Rota de Cobertura, CPP, Problema do Caixeiro Viajante Generalizado, GTSP, Otimização por Colônias de Formigas, ACO

# ABSTRACT

With the popularization of Unmanned Aerial Vehicles (UAVs), and with the expansion of the application areas of this technology, there is an increase of research focused on flight route planning. This work starts from the use of UAVs in agricultural scenarios in the aerial imaging task. The objective is to present a Coverage Path Planning (CPP) method for multicopter UAVs and compare it with a solution already used in the market. The developed method considers scenarios with multiple terrains, and seeks to present a route proposal with optimization of the order of visitation of the terrain, minimizing the number of curves of the route and the total distance of the route. The presented method contains three main steps. The first deals with the decomposition of areas, where terrains represented by concave polygons are decomposed into smaller convex shaped subareas using a greedy algorithm. The second step calculates the flight direction that minimizes the number of course curves in each of the subareas, finding the direction of the highest polygon height to guide the direction of the round-trip movement pattern application, known in the literature by default boustrophedon. The third step deals with subarea visit order optimization, at this stage the scenario is modeled as a specialization of the Generalized Travelling Salesman Problem (GTSP), and to solve this problem we use the Ant Colony Optimization algorithm (ACO). The results obtained by the proposed method are compared with the solutions proposed by a route planning program already used by the market, the Mission Planner. To measure the efficiency of the solutions, two variables were considered: the total distance traveled and the number of route curves. Through the obtained results one can map the types of scenarios where the developed method can aggregate with the market.

**Keywords:** VANT, Precision Agriculture, Coverage Path Planning, CPP, Generalized Travelling Salesman Problem, GTSP, Ant Colony Optimization, ACO



## LISTA DE FIGURAS

1	Figura representando os dois tipos principais de VANTs comerciais . . . . .	22
2	VANT multirotor preparado para a tarefa de aerofotogrametria . . . . .	26
3	Exemplo de caso de área onde o movimento de <i>boustrophedon</i> gera excessivo número de curvas para cobrir um terreno . . . . .	27
4	Exemplo de caso onde o algoritmo inseriu como parte da rota proposta coordenadas geográficas não pertencentes a área de interesse . . . . .	28
5	Grafo representando um exemplo de cenário típico do Problema do Caixeiro Viajante . . . . .	31
6	Representação em grafos de um cenário de STSP ao lado de um cenário de ATSP	32
7	Grafo representando umas das possíveis soluções para um cenário de GTSP com 6 grupos . . . . .	33
8	Número de trabalhos que abordam planejamento de rota de VANTs em fontes de informações acadêmicas, por ano de publicação . . . . .	35
9	Representação gráfica dos dois padrões básicos de preenchimento de área . . .	37
10	Representação gráfica do método decomposição celular exata . . . . .	38
11	Representação gráfica do método decomposição celular aproximada . . . . .	38
12	Comparação de uma representação gráfica e uma representação em grafo de um cenário com sete subáreas . . . . .	39
13	Representação gráfica de uma rota suavizada . . . . .	42
14	Representação gráfica do comportamento das formigas quando um obstáculo é inserido em uma trilha . . . . .	45

15	Representação gráfica de uma cenário com terrenos selecionados no <i>Google Earth</i> para serem processadas . . . . .	51
16	Cenário resultante do processo de decomposição das áreas de entrada . . . . .	52
17	Representações gráficas do comportamento do algoritmo de decomposição . . . . .	53
18	Representação gráfica das comparações de distância realizadas entre cada vértice e aresta do polígono que acontecem no algoritmo de cálculo de direção de voo. . . . .	55
19	Quatro alternativas de rotas diferentes que minimizam o número de curvas do trajeto para um dado polígono . . . . .	56
20	Representação gráfica de rotas internas resultantes do cálculo da orientação de voo . . . . .	57
21	Representação gráfica do modelo GTSP onde a rota escolhida deve partir e retornar para o mesmo nó, visitando somente um nó de cada terreno . . . . .	59
22	Cenário de teste para determinação dos parâmetros de controle do algoritmo de ACO . . . . .	62
23	Resultado do algoritmo ACO para o cenário de teste com utilizando os parâmetros ajustados . . . . .	63
24	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 1 . . . . .	67
25	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 2 . . . . .	70
26	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 3 . . . . .	72
27	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 4 . . . . .	74
28	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 5 . . . . .	76
29	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 6 . . . . .	78
30	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 7 . . . . .	80

31	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 8 . . . . .	82
32	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 9 . . . . .	84
33	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 10 . . . . .	86
34	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 11 . . . . .	88
35	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 12 . . . . .	91
36	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 13 . . . . .	93
37	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 13 . . . . .	95
38	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 14 . . . . .	97
39	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 15 . . . . .	99
40	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 16 . . . . .	101
41	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 18 . . . . .	103
42	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 19 . . . . .	105
43	Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 20 . . . . .	107

## LISTA DE TABELAS

1	Tabela resumindo o regulamento da ANAC para os diferentes tipos de aeronaves pilotadas remotamente . . . . .	24
2	Tabela com os valores e variáveis utilizados para gerar os lista de 625 combinações de parâmetros de controle do algoritmo ACO . . . . .	61
3	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 1 . . . . .	68
4	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 2 . . . . .	71
5	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 3 . . . . .	73
6	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 4 . . . . .	75
7	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 5 . . . . .	77
8	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 6 . . . . .	80
9	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 7 . . . . .	82
10	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 8 . . . . .	84
11	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 9 . . . . .	85

12	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 10 . . . . .	87
13	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 11 . . . . .	90
14	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 12 . . . . .	92
15	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 13 . . . . .	94
16	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 14 . . . . .	96
17	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 15 . . . . .	98
18	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 16 . . . . .	100
19	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 17 . . . . .	102
20	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 18 . . . . .	104
21	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 19 . . . . .	106
22	Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 20 . . . . .	108

## **LISTA DE SIGLAS**

**ACO** – *Ant Colony Optimization*

**ANAC** – *Agência Nacional de Aviação Civil*

**AP** – *Agricultura de Precisão*

**BVLOS** – *Beyond Visual Line Of Sight*

**CPP** – *Coverage Path Planning*

**EMBRAPA** – *Empresa Brasileira de Pesquisa Agropecuária*

**EVLOS** – *Extended Visual Line Of Sight*

**GCS** – *Ground Control Station*

**GIS** – *Geographic Information System*

**GPS** – *Global Positioning System*

**GSD** – *Ground Sample Distance*

**GTSP** – *Generalized Travelling Salesman Problem*

**IA** – *Inteligência Artificial*

**IMU** – *Inertial Measurement Unit*

**ISO** – *International Standards Organization*

**MAPA** – *Ministério da Agricultura Pecuária e Abastecimento*

**MDS** – *Modelos de Superfície*

**MDT** – *Modelos de Terrenos*

**NP** – *Non-Deterministic Polynomial Time*

**ORB** – *Oriented FAST and Rotated BRIEF*

**PP** – *Path Planning*

**RPA** – *Remotely Piloted Aircraft*

**SIFT** – *Scale Invariant Feature Transform*

**SURF** – *Speeded Up Robust Features*

**TP** – *Trajectory Planning*

**TSP** – *Travelling Salesman Problem*

**VANT** – *Veículo Aéreo Não Tripulado*

**VLOS** – *Visual Line Of Sight*

# SUMÁRIO

<b>CAPÍTULO 1 – INTRODUÇÃO</b>	<b>16</b>
1.1 Contexto . . . . .	16
1.2 Objetivos . . . . .	18
1.3 Organização do Trabalho . . . . .	19
<b>CAPÍTULO 2 – CONCEITOS BÁSICOS</b>	<b>21</b>
2.1 VANTs - Veículo Aéreo Não Tripulado . . . . .	21
2.2 Aerofotogrametria aplicada à agricultura de precisão . . . . .	24
2.3 Problema do Caixeiro Viajante e variantes . . . . .	30
<b>CAPÍTULO 3 – REVISÃO BIBLIOGRÁFICA</b>	<b>34</b>
3.1 Algoritmos de planejamento de rota . . . . .	34
3.2 Otimização por colônia de formigas . . . . .	44
<b>CAPÍTULO 4 – MÉTODO PROPOSTO</b>	<b>48</b>
4.1 Entrada de dados do sistema . . . . .	50
4.2 Subproblema 1 – Decomposição de polígonos . . . . .	51
4.3 Subproblema 2 – Navegação interna das subáreas . . . . .	54
4.4 Subproblema 3 – Otimização da ordem de visitação das subáreas . . . . .	57
4.5 Proposta de validação e de análise de resultados . . . . .	63
<b>CAPÍTULO 5 – RESULTADOS E ANÁLISES</b>	<b>65</b>



5.1	Cenários de terreno único: formato convexo . . . . .	66
5.1.1	Cenário 1 . . . . .	66
5.1.2	Cenário 2 . . . . .	69
5.1.3	Cenário 3 . . . . .	71
5.1.4	Cenário 4 . . . . .	73
5.1.5	Cenário 5 . . . . .	75
5.2	Cenários de terreno único: formato côncavo . . . . .	77
5.2.1	Cenário 6 . . . . .	77
5.2.2	Cenário 7 . . . . .	80
5.2.3	Cenário 8 . . . . .	82
5.2.4	Cenário 9 . . . . .	84
5.2.5	Cenário 10 . . . . .	85
5.3	Cenários com cinco terrenos . . . . .	87
5.3.1	Cenário 11 . . . . .	87
5.3.2	Cenário 12 . . . . .	90
5.3.3	Cenário 13 . . . . .	92
5.3.4	Cenário 14 . . . . .	94
5.3.5	Cenário 15 . . . . .	96
5.4	Cenários com dez terrenos . . . . .	99
5.4.1	Cenário 16 . . . . .	99
5.4.2	Cenário 17 . . . . .	101
5.4.3	Cenário 18 . . . . .	103
5.4.4	Cenário 19 . . . . .	105
5.4.5	Cenário 20 . . . . .	106
5.5	Conclusão . . . . .	109

# Capítulo 1

## INTRODUÇÃO

---

---

### 1.1 Contexto

O avanço da tecnologia nas mais diversas áreas de aplicações é uma realidade que vivemos. Devido ao seu poder de reduzir os custos de processos e aumentar a produtividade, cada vez mais nota-se indústrias com linhas de montagem automatizadas, utilizando tecnologias de ponta, robôs e computadores, com o objetivo de aumentar a eficiência dos processos. Com a agricultura não é diferente. A preocupação em aumentar a produtividade e diminuir os riscos ao ambiente e às pessoas fez com que a área necessitasse da inserção de tecnologia para atingir estes objetivos, possibilitando assim uma maior variabilidade e precisão no manejo das culturas, dando início à chamada Agricultura de Precisão (AP).

O Ministério da Agricultura, Pecuária e Abastecimento (MAPA), define Agricultura de Precisão como sendo “um sistema de gerenciamento agrícola baseada na variação espacial e temporal da unidade produtiva e visa ao aumento de retorno econômico, à sustentabilidade e à minimização do efeito ao ambiente” (BRASIL, 2012).

Essa inserção da tecnologia no meio agrícola aconteceu de modo gradativo. Inamasu e Bernardi (2014) citaram que na década de 80 a disseminação da eletrônica embarcada impulsionou o aparecimento de máquinas programáveis, veículos autoguiados e robôs industriais. Já na década de 90 com a agricultura de precisão ganhando força, muitas das tecnologias e estudos do meio industrial migraram para a área agrícola. Com a popularização da tecnologia de GPS, as rotas dos veículos autônomos ganharam maior flexibilidade, passando de simples robôs seguidores de linhas para máquinas que conseguem processar informações georreferenciadas e locomoverem-se sem a interferência do piloto.

Atualmente os produtores contam com uma série de equipamentos que auxiliam na ro-

tina agrícola: maquinários e pulverizadoras com sistemas de controle individual dos bicos de aplicação, que possibilitam um manejo variado e mais preciso; sistema de posicionamento global (*Global Positioning System* - GPS) e piloto automático; sensores hiperespectrais e infravermelhos com capacidade de obter dados do campo além do espectro visível; aeronaves e Veículos Aéreos Não Tripulados (VANTs) para tarefas de monitoramento, pulverização de culturas e geração de mapas de características do solo (INAMASU; BERNARDI, 2014).

Estas tecnologias agrícolas chegaram ao campo trazendo não só facilidades, mas também desafios. O uso de máquinas e VANTs autônomos para realização de tarefas nas áreas de cultivos está atrelado a elaboração de um planejamento de rota. O objetivo do planejamento de rota é indicar um caminho mínimo para o veículo percorrer e cumprir sua tarefa de forma segura, eficiente, e gastando a menor quantidade de recursos possíveis (DE ALMEIDA *et al.*, 2013).

Como citado no trabalho de Oksanen e Visala (2007) o planejamento de rota (*Path Planning* - PP) no meio agrícola possui diversos usos. Pode ser aplicado tanto nas máquinas autônomas, quanto na determinação de um caminho otimizado para que o agricultor possa dirigir suas máquinas de forma a economizar recursos. Além da aplicação para os maquinários terrestres, esta técnica também é utilizada para os aviões agrícolas e VANTS, estendendo ainda mais sua utilização. O planejamento de rota pode ser aplicado no meio agrícola em tarefas como o de plantação de sementes (BLENDER *et al.*, 2016), aplicação de pesticidas (LAL; SHARDA; PRABHAKAR, 2017), colheita (PLESSEN, 2019) entre outras mais, onde o agente necessita percorrer a cultura para realizar uma determinada tarefa.

Este trabalho foca em um tipo específico de algoritmo de rota, o planejamento de rota de cobertura (*Coverage Path Planning* - CPP). O CPP é um algoritmo que tem por objetivo definir uma rota mínima que percorra inteiramente uma área. Este tipo de algoritmo é utilizado em robôs para tarefas de limpeza (MIAO; LEE; KANG, 2018), de monitoramento (KHOUFI; MINET; ACHIR, 2016), de cobertura de incêndios (PHAM; BESTAOUI; MAMMAR, 2017) até mesmo em sistemas de pintura industriais (PIRES *et al.*, 2009).

Neste trabalho, a aplicabilidade do algoritmo de CPP tem foco no planejamento de rota para a tarefa de aerofotogrametria utilizando VANTs. Neste processo citado, são adquiridos dados topográficos através de imagens aéreas, capturadas com uso de câmeras acopladas nos VANTs. A intenção de capturar imagens no processo de aerofotogrametria é extrair informações que auxiliem o produtor agrícola a tomar decisões sobre onde e como atuar em sua lavoura. Apesar do foco na tarefa de aerofotogrametria, os algoritmos e técnicas estudadas neste trabalho são válidas e reaproveitáveis em várias outras aplicações já citadas.

A motivação de estudar a otimização de rota de cobertura parte da necessidade observada

tanto nos trabalhos recentes, como no de Wu *et al.* (2019) e Modares *et al.* (2017), quanto em experiências de campo, onde a bateria e a autonomia do drone tornam-se fatores críticos em frente à grande quantidade de terra a serem mapeadas. Desta forma, este trabalho visa propor um método de planejamento de rota de cobertura otimizado que contribua com as soluções já existentes atualmente para gerar propostas de rotas cada vez melhores buscando assim minimizar gastos de recursos desnecessários. O método deste trabalho será comparado com uma solução já aplicadas atualmente por profissionais da área, buscando identificar os cenários em que o algoritmo desenvolvido possa contribuir positivamente com as soluções já utilizadas, aumentando a eficiência das rotas propostas.

A literatura apresenta diversos trabalhos com estratégias, abordagens e técnicas diferentes que buscam o mesmo objetivo de apresentar uma rota otimizada e eficiente. A grande parte destas pesquisas procuram atingir esse resultado através da minimização da distância percorrida, ou seja, trabalham em cima da variável da distância buscando utilizar métodos que priorizem concluir uma determinada tarefa percorrendo a menor distância possível. O trabalho de Strimel e Veloso (2014) é um dos que seguem este tipo de abordagem.

Além de minimizar a distância percorrida, alguns trabalhos defendem que outras variáveis influenciam também nas solução de rota apresentadas, como é o caso de Coombes *et al.* (2018) que chama a atenção para o fato de que utilizar a direção do vento para traçar um plano de voo que minimize o tempo do VANT no ar contribui para o aumento da eficiência da solução. Já Vasquez-Gomez *et al.* (2016) utiliza a variável vento em seu modelo de uma maneira diferente, buscando tratá-la em seu modelo para poder orientar o plano de voo de forma que minimize o consumo de bateria, justificando que nem sempre a menor rota é a mais econômica. Nos trabalhos de Li *et al.* (2011) e Torres *et al.* (2016), o planejamento de rota estudados tinha o objetivo de minimizar o número de curvas do trajeto, partindo do fato de que curvas causam um gasto maior de recursos.

As soluções multivariáveis, ou seja, as que buscam otimizar a rota levando em conta mais de uma variável em seu modelo, costumam apresentar resultados melhores que os sistema que consideram uma única variável, uma vez que o modelo estudado tende a se aproximar mais do cenário real.

## **1.2 Objetivos**

Este trabalho tem como objetivo principal apresentar um método de planejamento de rotas de cobertura otimizado, aplicado a coleta de imagens em terrenos agrícolas com VANT multi-

rotor.

O método proposto se baseia na aplicação de Inteligência Artificial (IA) no processo de planejamento de rota, mais precisamente no uso de uma técnica de meta-heurística conhecida por otimização de colônia de formigas (*Ant Colony Optimization* - ACO), para assim propor rotas otimizadas em cenários com múltiplos terrenos. Para refletir cenários reais, além de tratar cenários com múltiplos terrenos, o método deve aceitar tanto terrenos de formato convexo quanto côncavos.

Com base em um cenário de entrada apresentado, o método desenvolvido deve apresentar uma sugestão de rota de cobertura que otimize ordem de visitação das áreas, minimize o número de curvas do percurso e a distância total percorrida. As rotas apresentadas serão comparadas com rotas do *Mission Planner*, programa de planejamento de rota já utilizado pelos profissionais da área.

O processo de comparação e validação de resultados é um dos aspectos onde este trabalho busca agregar ao estado da arte. A maioria das pesquisas encontradas que aplicam algum método de planejamento de rota em cenários agrícolas, se baseiam em um estudo de caso particular, utilizando suas próprias métricas para avaliar a eficiência da solução. Características como estas dificultam a comparação das soluções com outros trabalhos, principalmente pela falta de disponibilidade dos dados de testes. Deste modo é também um objetivo deste trabalho disponibilizar dados suficientes para que os testes possam ser reproduzidos e comparados. Para a comparação de resultados, as rotas propostas pelo método desenvolvido serão confrontadas com as soluções propostas por um programa de planejamento de rotas largamente utilizado no meio profissional, o *Mission Planner*. Serão utilizados múltiplos cenários de aspectos diferentes, procurando identificar em quais cenários o método consegue resultados positivos e em quais cenários o método desenvolvido não é indicado. Esta variação de cenários de aspectos diferentes contribui para a criação de um banco de casos que podem servir de comparação para outros trabalhos. O dados de testes utilizados serão disponibilizados para acesso público, buscando contribuir com futuros trabalhos.

## 1.3 Organização do Trabalho

Para a escrita deste trabalho, o documento foi organizado em cinco capítulos. O capítulo 1 contextualiza o leitor sobre a área de pesquisa do trabalho, trazendo o problema da área que será abordado, seguido do objetivo do estudo. O capítulo 2 apresenta conceitos básicos relacionado ao tema de pesquisa, passando por VANTs, aerofotogrametria e o Problema do Caixeiro Via-

jante. No capítulo 3 se encontra a revisão bibliográfica, trazendo principalmente informações sobre trabalhos com foco no planejamento de rota. O capítulo 4 registra a proposta do trabalho com base nas informações levantadas na revisão bibliográfica. O capítulo 5 é composto pelos resultados, análises e conclusão.

# Capítulo 2

## CONCEITOS BÁSICOS

---

---

Por tratar-se de uma pesquisa multidisciplinar, notou-se a necessidade de primeiramente apresentar alguns conceitos básicos relacionados com a área de estudo, para deste modo apresentar ao leitor o escopo da pesquisa. A seção 2.1 tem como tema os Veículos Aéreos Não Tripulados (VANTs) e sua legislação no Brasil. A seção 2.2 trata do uso dos VANTs na tarefa de aerofotogrametria com foco na agricultura de precisão. Na seção 3.1 são apresentados conceitos do Problema do Caixeiro Viajante e algumas de suas variantes.

### **2.1 VANTs - Veículo Aéreo Não Tripulado**

Na década de 80 surgiram os primeiros VANTs no Brasil. No início eram utilizados na área militar para substituir pessoas em operações arriscadas. Com o passar do tempo a tecnologia dos VANTs foi se tornando mais acessível, com isso seu uso foi estendido para as mais diversas áreas, inclusive na agricultura de precisão. Jorge e Inamasu (2014) citaram que os VANTs, com o objetivo de diminuir os gastos de recursos e aumentar a eficiência da produção agrícola, passaram a realizar tarefas de monitoramento e mapeamento de culturas para detecção de falhas em plantio, controle de pragas, aquisições de imagens aéreas para geração de modelos de superfície (MDS), modelos de terreno (MDT) e ortomosaicos, sendo até utilizados em tarefas de pulverização, como é o caso do trabalho de Sinha (2017) que busca melhorar a eficiência no processo de aplicação de pesticida em culturas com o uso de VANTs.

Comercialmente os VANTs são divididos em dois tipos principais: asa fixa e multirotor, como mostrado na Figura 1.

**Figura 1: Figura representando os dois tipos principais de VANTs comerciais**

(a) VANT do tipo asa fixa



(b) VANT do tipo multirotor

**Fonte: Disponível em: <https://horusaeronaves.com> e <https://www.dji.com>, respectivamente.**

**Acesso em: julho de 2018**

Boon, Drijfhout e Tesfamichael (2017) fizeram um estudo comparativo, realizado com base em um caso real de levantamento topográfico, onde apresentaram um conjunto de prós e contras dos principais modelos de VANTs. Os VANTs do tipo asa fixa, em geral, possuem uma maior autonomia de voo comparado aos multirotores. O modelo Verok, de acordo com seus fabricantes, tem autonomia de duas horas de voo. Um ponto negativo dos VANTs de modelo asa fixa é a sua necessidade de um espaço livre e amplo para decolagem e aterrissagem, já que são realizadas no sentido horizontal. Por sua vez, VANTs do tipo multirotor, além da capacidade de pararem no ar, possuem a vantagem de ter sua decolagem e pouso no sentido vertical, o que facilita seu uso nos mais diversos cenários. Por outro lado sua grande desvantagem é a autonomia de voo que é mais baixa comparado ao modelo asa fixa. O multirotor Phantom 4 Pro, de acordo com seus fabricantes, tem autonomia de 30 minutos. Os modelos multirotores possuem também a vantagem de utilizar mais facilmente estabilizadores de câmeras, as chamadas *gimbals*, que diminuem os tremores da câmera durante o voo, mantendo-as fixas em uma determinada orientação independente do movimento do VANT.

Cada tipo de VANT possui um conjunto de particularidades que deve ser considerado na modelagem do sistema a ser estudado. Bulusek (2015), por exemplo, realizou uma pesquisa com o intuito de apresentar uma alternativa de planejamento de rota de cobertura otimizada com foco na aplicação de aerofotogrametria, e o modelo de VANT utilizado em seu trabalho foi um asa fixa, o que influenciou em seus cálculos de rota, principalmente nas curvas, por conta da inclinação inerente do modelo utilizado.

Com o objetivo de aumentar a segurança das pessoas e tornar viável as diversas aplicações utilizando VANTs, a Agencia Nacional de Aviação Civil (ANAC), criou uma conjunto de regras para operações utilizando VANTs baseado nas legislações de outros países, como Estados Unidos e países da União Europeia.



Essa legislação entrou em vigor em 2 maio de 2017, e nela a ANAC divide os VANTS em dois grupos baseados na finalidade das aeronaves, sendo elas: Aeromodelos, definidos como aeronaves para fins recreativos; Aeronaves Remotamente Pilotados (*Remotely Piloted Aircraft* - RPA), aeronaves utilizadas em atividades corporativas, comerciais e experimentais (ANAC, 2017). O termo drone também é citado no regulamento, mas de uma forma mais geral e popular fazendo referencia a qualquer modelo de aeronave não tripulada.

ANAC (2017) subdivide as operações utilizando aeronaves em 3 grupos:

- VLOS (*Visual Line Of Sight*) - operação em que o piloto mantém contato direto com a aeronave.
- BVLOS (*Beyond Visual Line Of Sight*) - operação em que o piloto não tem nenhum tipo de contato visual direto com a aeronave.
- EVLOS (*Extended Visual Line Of Sight*) - operação em que o piloto mantém contato visual com a aeronave com o auxílio de lentes, equipamento e observadores.

Além das operações, o regulamento aprovado pela ANAC, subdivide as aeronaves do tipo RPA em três classes:

- classe 1 - aeronaves acima de 150Kg.
- classe 2 - aeronaves acima de 25 Kg e abaixo ou igual a 150 kg.
- classe 3 - aeronaves abaixo ou igual a 25 Kg.

A legislação é diferentes para cada classe de aeronave, quanto maior o peso da aeronave, maiores são as exigências para pilotá-las. No regulamento fica proibido o uso aeronaves totalmente autônomas, ou seja, o piloto responsável deve conseguir assumir o controle manual a qualquer momento.

Na classe dos aeromodelos, por se tratar de uso recreativo, as leis são menos rígidas. Aero-modelos com mais de 250g devem ser cadastrados na ANAC, o piloto pode ter qualquer idade e deve operar em uma distância de 30 metros de terceiros.

Já para as aeronaves classificadas como RPA (*Remotely Piloted Aircraft* - RPA), a legislação muda de acordo com sua classe. Para ser operada, poderá ser exigido uma autorização de voo da ANAC, habilitação do piloto, certificado médico aeronáutico, e que o voo seja registrado.

Segue abaixo um tabela adaptada do regulamento ANAC (2017) contendo o resumo das regras de voos para cada categoria.

**Tabela 1: Tabela resumindo o regulamento da ANAC para os diferentes tipos de aeronaves pilotadas remotamente**

	RPA - Classe 1	RPA - Classe 2	RPA - Classe 3	Aeromodelo
Registro da aeronave?	Sim	Sim	BVLOS: Sim VLOS: Sim	Sim, se possui mais que 250g
Aprovação ou autorização do projeto?	Sim	Sim	Apenas BVLOS ou acima de 120 metros	Não
Limite de idade para operação?	Sim	Sim	Sim	Não
Certificado médico?	Sim	Sim	Não	Não
Licença e habilitação?	Sim	Sim	Apenas para operações acima de 120 metros	Apenas para operações acima de 120 metros
Local de operação	A distância da aeronave não pode ser inferior a 30 metros horizontais de pessoas não envolvidas na operação			

**Fonte: Adaptado de ANAC (2017)**

Está registrado no regulamento que pessoas, animais e substâncias perigosas não podem ser transportados por VANTs, mas quanto se tratar de substâncias para lançamento em atividades de agricultura, horticultura ou florestais o transporte é permitido. Também é permitido o transporte de baterias de lítio e equipamentos eletrônicos, como câmeras fotográficas, caso sejam utilizados durante o voo.

Com base no regulamento da ANAC nota-se que para utilizar um VANT de classe 3 para as aplicações agrícolas é necessário que: a carga total do VANT seja até 25 kg, a altura de voo seja até 120 metros, o VANT seja registrado, e o piloto seja maior de idade e tome as devidas precauções de segurança. Seguindo estas regulamentações pode-se por exemplo equipar o VANT com câmeras fotográficas e utilizá-lo para atividades agrícolas de aerofotogrametria.

## 2.2 Aerofotogrametria aplicada à agricultura de precisão

A aerofotogrametria, como dito anteriormente, é o levantamento topográfico realizado através de imagens aéreas de onde pode-se obter medidas reais dos elementos das imagens. A aerofotogrametria é um processo relativamente antigo, em meados do século XVIII já haviam registros de fotos tiradas a bordo de balões com objetivo de criar mapas, processo este que posteriormente passou a ser realizado com aeronaves, satélites e atualmente também está sendo realizado com VANTs (BULUSEK, 2015). Uma das vantagens da aerofotogrametria utilizando VANTs sobre as que utilizam satélites é a resolução das imagens obtidas. As imagens tiradas de VANTs conseguem resolução de 10 centímetros ou menos, dependendo da altura de voo e câmera uti-

lizada, contra em média 50 centímetros das imagens de satélites. A técnica do imageamento aéreo utilizando VANTs evita o problema de existirem nuvens entre o sensor e o solo, situação que é comum nas imagens aéreas obtidas por satélites (JORGE; INAMASU, 2014). Outra vantagem é que aerofotogrametria realizada por VANTs pode acontecer em períodos desejados pelo agricultor, portanto podem ser realizados múltiplos voos acompanhando as diferentes fases da cultura. O uso desta técnica utilizando VANTs gera um leque de ferramentas muito úteis para a agricultura de precisão como ortomosaicos, modelos de superfície (MDS), modelos de terrenos (MDT) e índices vegetativos (EISENBEI, 2009). Para o produtor agrícola ou até mesmo sua equipe de agrônomos, essas ferramentas são de grande importância para que o ciclo de observação, planejamento e tomada de ação ocorra de forma mais rápida e eficaz.

O conjunto de um VANT equipado com uma unidade de navegação inercial (*Inertial Measurement Unit* - IMU), um sistema de posicionamento global (*Global Positioning System* - GPS) e uma câmera fotográfica formam o conjunto de ferramentas básicas para o uso do VANT na aerofotogrametria como é retratado na Figura 2. É necessário também uma estação de controle de voo, também chamada de GCS (*Ground Control Station*), que é composta por um conjunto de *hardware* (computador e telemetria) e *software* (programas para planejamento de voos). São exemplos de programas de planejamento de voo, *Mission Planner*, *MAVProxy*, *QgroundControl*, *UgCS*, *Tower* e *MAVPilot*. Nas estações de controle além da determinação do plano de voo do VANT é possível também realizar todo o acompanhamento do voo em tempo real, monitorando variáveis como altitude, nível da bateria, inclinação do VANT e direção do vento. Para o estudo em questão, foi escolhido o *Mission Planner* como o programa de planejamento de rotas para serem realizadas as comparações com o método desenvolvido neste trabalho, pelo motivo do *Mission Planner* de ser programa de planejamento de rota completo, amplamente utilizado no meio profissional dos pilotos de VANTs, e ter seu código aberto e disponível para alterações.

**Figura 2: VANT multirotor preparado para a tarefa de aerofotogrametria**

**Fonte: Imagem cedida pela equipe de drones para agricultura de precisão da EMBRAPA**

Resumidamente, para realizar o processo de coleta de imagens, o piloto no comando da estação de controle configura o VANT para sobrevoar uma área desejada e capturar imagens. Para o planejamento da rota, de acordo com o manual do *Mission Planner*, deve ser fornecido ao programa de planejamento de rota um polígono que represente a área em 2D a ser mapeada, a altura do voo, o percentual desejado de sobreposição das imagens e as características do sensor da câmera. Com base nas informações de entrada, usando o movimento de *boustrophedon* (movimento de ida e vinda), é gerada automaticamente uma rota de cobertura da área. Este procedimento vem sendo utilizado por profissionais da área de aerofotogrametria e na grande maioria dos casos cumpre seu objetivo com eficiência.

O problema ocorre quando as áreas a serem mapeadas não são regulares e possuem uma certa complexidade, nestes casos as rotas de cobertura geradas automaticamente podem ser ineficientes, como é o caso da Figura 3, pois como pode-se observar na parte esquerda do terreno, ocorreram movimentos desnecessários para cobrir a área. Esta rota apresenta um excessivo número de curvas quando aplicado o movimento de *boustrophedon* em um mesmo sentido para a área inteira. Portanto utilizar uma orientação de voo inadequada, colabora para um uso ineficiente de recursos.

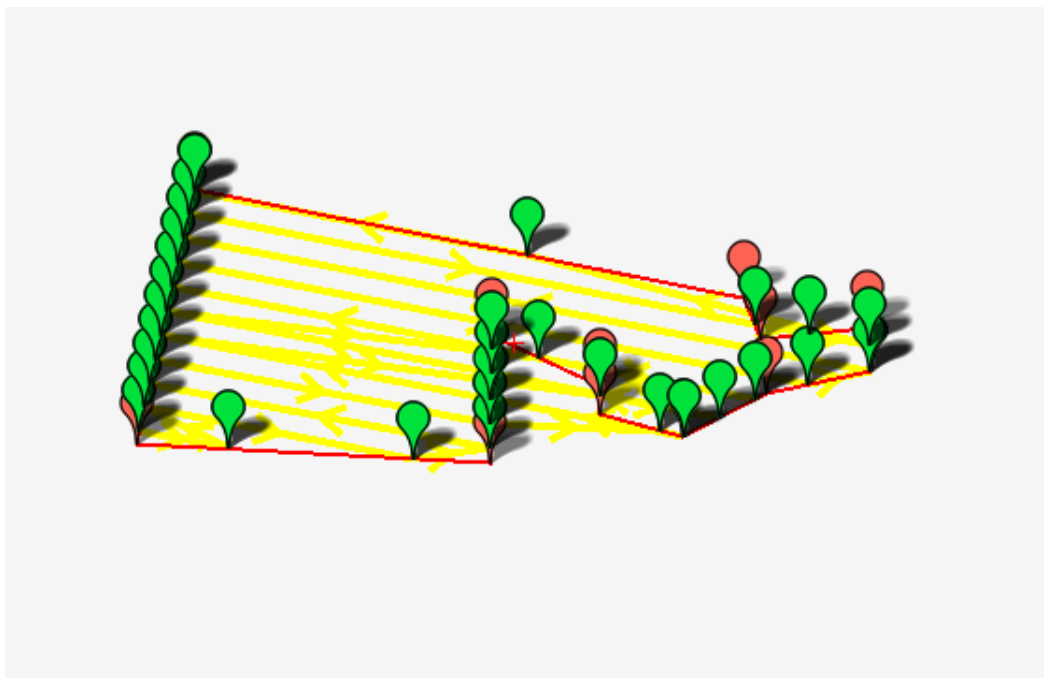
**Figura 3:** Exemplo de caso de área onde o movimento de *boustrophedon* gera excessivo número de curvas para cobrir um terreno



Fonte: Própria (2019)

Outro problema aparece quando a solução proposta pelo algoritmo engloba sobrevoar uma área que não é de interesse da aplicação, ou seja, para atingir o mapeamento completo da área de interesse, parte da rota proposta acontece fora da região de interesse, o que pode resultar em desperdício de recursos dependendo do cenário. Essa situação é retratada na Figura 4

**Figura 4:** Exemplo de caso onde o algoritmo inseriu como parte da rota proposta coordenadas geográficas não pertencentes a área de interesse



**Fonte:** Própria (2019)

De forma mais detalhada, Jorge e Inamasu (2014) listaram os passos da fotogrametria aplicada à agricultura da seguinte forma: planejamento de voo (com sobreposição de imagens), obtenção de imagens georreferenciadas, processamento das imagens para a geração de um mosaico, análise do mosaico em ferramentas de informação geográfica (*Geographic Information System* - GIS) para obtenção de informações relevantes sobre a área sobrevoada.

Na etapa do planejamento de voo acontece a parametrização de algumas variáveis, como: velocidade de voo, área a ser sobrevoada, altitude de voo, taxa de sobreposição de imagens, valor de distância amostral do solo (*Ground Sample Distance* - GSD), que é a medida de quantos centímetros do solo será representado em um pixel da imagem. Cada uma dessas variáveis deve ser estipulada de acordo com o objetivo do voo e dos equipamentos que serão utilizados. O valor da altura do voo deve ser calculado levando em conta os parâmetros intrínsecos da câmera que será utilizada e a resolução do pixel desejada na aplicação.

Existe um amplo leque de câmeras que podem ser utilizadas para a tarefa de obtenção de imagens aéreas como por exemplo, câmeras no espectro visível, infravermelho, multiespectral, hiperespectral, sendo que cada uma atua em uma faixa do comprimento de onda, portanto cada uma obtêm informações diferentes do ambiente. O trabalho de Jorge e Inamasu (2014) cobre com mais detalhes as principais diferenças entre as câmeras.

Na fase de obtenção de imagens georreferenciadas, é necessário que exista sobreposição entre as imagens, ou seja, cada nova imagem deve registrar também parte do conteúdo da anterior, pois somente deste modo será possível utilizar algoritmos para identificar características em comum nas diferentes fotos e juntá-las em uma única imagem da área total sobrevoada. Esta imagem resultante recebe o nome de ortomosaico. Para VANTs a taxa de sobreposição comumente utilizada fica dentro da faixa de 40% a 80% (JORGE; INAMASU, 2014). É necessário também, nesta fase de captura, que exista um sistema de controle de disparo sequencial da câmera e que o VANT se mantenha em uma velocidade constante para que as imagens sejam capturadas cadencialmente, já que a velocidade de voo e o tempo de disparo estão estritamente ligados em função de manter a taxa de sobreposição definida. Nos metadados de cada imagem capturada são gravados valores de latitude, longitude, variáveis inerciais, entre outras, para que assim cada imagem possa ser reajustada, e georreferenciada no momento da criação de mosaico. Jorge e Inamasu (2014) citaram que alguns parâmetros de câmera como exposição de branco, valor de ISO, ajuste de foco e velocidade do obturador influenciam muito na qualidade da imagem capturada e conseqüentemente no resultado final do mosaico, portanto estes parâmetros devem sempre serem ajustados antes de cada voo. Na fase seguinte a captura, as imagens são processadas e o mosaico é construído.

O mosaico, ou também chamado ortomosaico, é a junção de todas as imagens capturadas durante o voo sobre uma área de interesse, que posteriormente são processadas para compor uma única imagem equivalente a área total, se tornando um produto necessário para várias análises agrícolas posteriores. A criação de um ortomosaico por si só é um processo complexo e amplamente estudado. Trabalhos como o de Yahyanejad (2013) trazem informações detalhadas sobre técnicas de construções de mosaicos. Resumindo o processo de construção de um mosaico, cada imagem passa por um reajuste, onde são corrigidos angulações de captura, e posteriormente são identificados *features* (características singulares) em cada imagem com o uso de algoritmos identificadores de características, como *SIFT*, *SURF* e *ORB*. Devido à taxa de sobreposição entre as imagens é possível identificar uma mesma característica singular nas imagens vizinhas, e esse processo de parear *features* entre as imagens acontece na fase conhecida na literatura por *match*. Com base nas junções das imagens, ocorre um cálculo de predição das posições da câmera no momento do disparo, definindo assim valores extrínsecos iniciais para cada imagem, como rotação e translação. Na fase seguinte ocorrem cálculos para refinar os valores extrínsecos da câmera, corrigindo conseqüentemente as posições de GPS de cada imagem para que assim, as junções das imagens fiquem mais precisas, esta fase é conhecida na literatura por *bundle adjustment*. Na fase de junção são aplicadas técnicas de suavização de textura, ou também conhecida por *blending* para eliminar emendas entre as imagens. No final

do processo obtêm-se o ortomosaico, onde pode ser utilizado escalas para retiradas medidas e informações confiáveis sobre a área de interesse (BULUSEK, 2015).

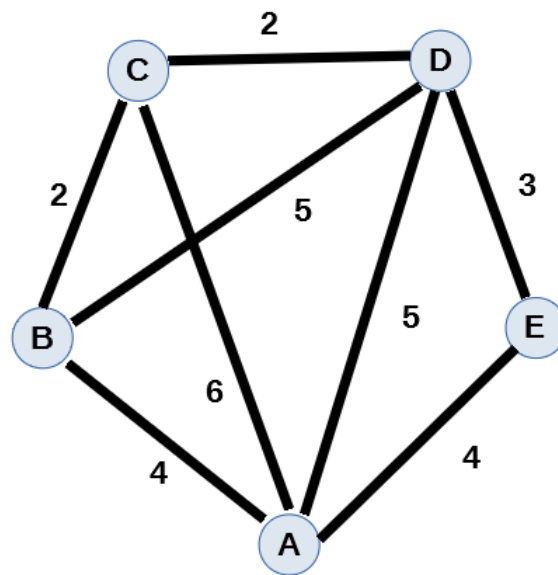
Através da aplicação de índices vegetativos sobre o mosaico e nas imagens processadas, é possível obter informações úteis ao agrônomo, possibilitando que ações possam ser tomadas como, corrigir falhas de plantio ou direcionar a aplicação de pesticidas de uma maneira localizada e mais eficiente. A tecnologia dos VANTs aplicados a agricultura de precisão, vem tornando o processo produtivo mais eficaz, mas ainda contam com o fator crítico de baixa eficiência frente à grande quantidade de terrenos a serem analisados. Este é um dos motivos que torna a otimização do consumo de bateria um objeto de estudo constante, já que aumentar a eficiência do uso da bateria resulta em aumentar o tempo de voo e a quantidade de área sobrevoada.

Atualmente são diversas áreas de pesquisas que buscam melhorar o desempenho e a eficiência dos VANTs em voo. Estudos como o de Gandolfo *et al.* (2017) buscam entender, testar e avaliar os gastos de energia de diferentes sistemas de controle de VANTs. Já neste trabalho, o foco é otimizar a rota de voo, para que o VANT cumpra seu objetivo de sobrevoar completamente uma área de uma maneira mais eficiente, evitando passar por caminhos já visitados, e fazendo o mínimo de movimentos desnecessários para não desperdiçar energia.

## 2.3 Problema do Caixeiro Viajante e variantes

Um dos problemas tratado no método proposto deste trabalho, pode ser modelado de forma similar ao Problema do Caixeiro Viajante (*Travelling Salesman Problem - TSP*). Este problema possui diversas variações. Bonyadi, Azghadi e Shah-Hosseini (2008) citaram em seu trabalho que o modelo primário do TSP pode ser descrito utilizando o seguinte cenário: um vendedor recebe um mapa contendo diversas cidades, seu objetivo é visitar todas as cidades passando por cada uma delas somente uma vez, de maneira que percorra a menor distância possível. As distâncias entre cada cidades são descritas no mapa como ilustra a Figura 5. Nesta figura os pontos com letras representam as cidades à serem visitadas, e os números marcam as distâncias entre elas. Neste caso, o vendedor partindo da cidade A, por exemplo, deve retornar a ela depois de visitar as cidades B, C, D e E, em qualquer ordem, passando somente uma vez em cada, de forma a percorrer a menor distância possível.

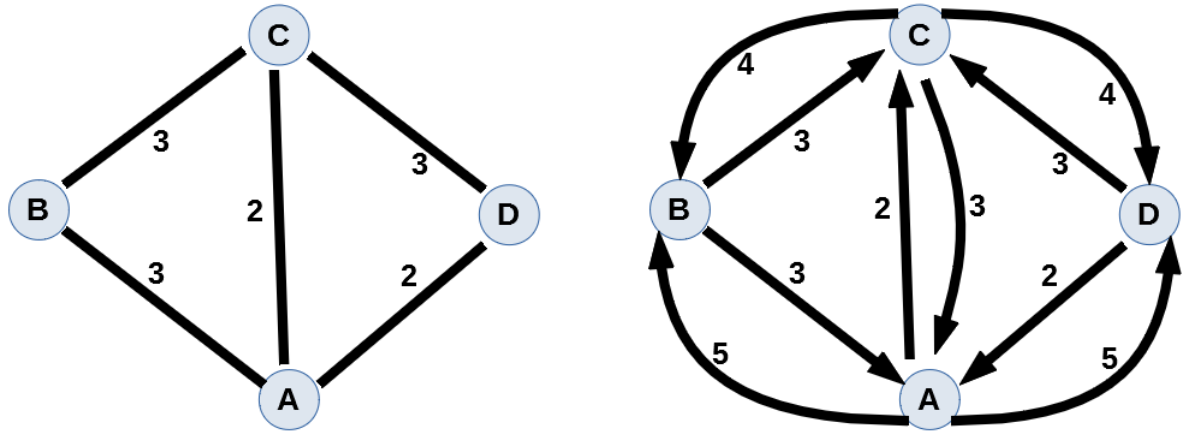


**Figura 5: Grafo representando um exemplo de cenário típico do Problema do Caixeiro Viajante**

Fonte: Própria (2019)

Este problema pode ser de simples resolução quando se trata de um cenário com um número pequeno de cidades. Por outro lado, o Problema do Caixeiro Viajante tem complexidade de resolução alta para um número grande de cidades, podendo demorar anos para atingir uma solução computacional se forem testados todas as possíveis alternativas de rotas. O TSP é considerado um problema NP-Hard (HARTMANIS, 1982), e representa uma ampla classe de problemas de otimização combinatória.

Duas das variações mais populares do Problema do Caixeiro Viajante são: o Problema do Caixeiro Assimétrico (*Assimetric Travelling Salesman Problem - ATSP*) e o Problema do Caixeiro Viajante Simétrico (*Simetric Travelling Salesman Problem - STSP*). No modelo STSP as distâncias de ida e vinda entre as cidades são iguais, portanto na representação gráfica as cidades são interligadas com uma única linha, como representado na Figura 6(a). No ATSP as distâncias de ida e vinda se diferem entre pelo menos duas cidades do grafo, como ilustra a Figura 6(b), desta forma, é necessário que na representação gráfica as interligações entre as cidades sejam representadas por setas de ida e de vinda com suas respectivas distâncias (GUTIN; PUNNEN, 2006).

**Figura 6: Representação em grafos de um cenário de STSP ao lado de um cenário de ATSP**

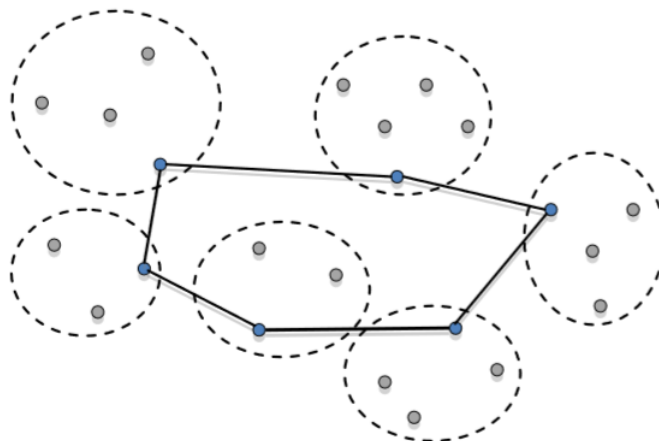
(a) Exemplo de cenário de um STSP

(b) Exemplo de cenário de um ATSP

Fonte: Própria (2019)

Parte do problema a ser resolvido neste trabalho, pode ser modelado como uma variante do TSP, denominada Problema do Caixeiro Viajante Generalizado (*Generalized Travelling Salesman Problem - GTSP*). Esta especialização se difere do TSP no sentido que cada cidade pertence a um grupo, e neste cenário não é necessário que todas as cidades sejam visitadas, bastando que na solução todos os grupos sejam visitados. A Figura 7 exemplifica um caso de GTSP. Sendo assim, no GTSP o objetivo é visitar todos os grupos, passando pelo menos por um nó de cada grupo (SILBERHOLZ; GOLDEN, 2007)

**Figura 7: Grafo representando umas das possíveis soluções para um cenário de GTSP com 6 grupos**



**Fonte: Adaptado de Helsgaun (2015)**

Trabalhos como a de Ben-Arieh *et al.* (2003) e Khachai e Neznakhina (2017) citaram o fato de que um problema do tipo GTSP pode ser resolvido através da conversão do modelo GTSP para um modelo do tipo TSP. A vantagem desta conversão é a possibilidade de se utilizar os métodos de resolução de problemas do tipo TSP, que são mais difundidos na literatura que resoluções de GTSP. Porém, uma das desvantagens deste método é que para conseguir que o problema seja modelado como TSP são necessários uma série de passos e manipulações matemáticas, além de resultar em um aumento do número de nós do grafo em relação ao cenário do problema inicial. Para adaptar um problema GTSP para TSP, primeiramente é preciso que o modelo seja convertido de GTSP para um modelo ATSP (*Asymmetric Travelling Salesman Problem*). Uma das forma de realizar esta primeira conversão é através de um algoritmo proposto por Noon e Bean (1993). Somente depois desta conversão é que o modelo ATSP poderá ser convertido para um modelo TSP. Outro método de resolução possível é através dos algoritmos de rede como citou Khan, Noreen e Habib (2017).

Em casos em que para atingir a solução de um cenário de GTSP o caixeiro deva passar somente por uma cidade de cada grupo, o problema recebe o nome E-GTSP (*Equality Generalized Travelling Salesman Problem*) (HELSGAUN, 2015). Este tipo de cenário é exatamente o problema que deve ser resolvido na terceira parte do algoritmo de planejamento de rota proposto portanto será tratada com mais detalhes no Capítulo 4.

# Capítulo 3

## REVISÃO BIBLIOGRÁFICA

---

---

Neste capítulo são abordados os trabalhos resultantes da revisão sistemática que trouxeram informações relevantes para a pesquisa em questão, mais precisamente relacionados aos temas de algoritmos de planejamento de rota e otimização por colônia de formigas.

### 3.1 Algoritmos de planejamento de rota

Como citado anteriormente no Capítulo 1 este trabalho busca utilizar algoritmos de planejamento de rota para contribuir com a minimização do gastos de recursos. Existe um campo da robótica que estuda exclusivamente os algoritmos de planejamento de rotas de robôs autônomos. Baseados em conceitos dessa área, primeiramente, será esclarecida a diferença entre planejamento de trajetória e planejamento de rota, pois são termos relacionados, mas com significados diferentes. Coutinho, Battarra e Fliege (2018) citaram o planejamento de trajetórias, ou *Trajectory Planning* como um controle de manobra, envolvendo cinemática, posição, velocidade, aceleração e considerando também forças aplicadas sobre o corpo do VANT, desta forma relacionando a variável espaço com a variável tempo. Exemplos de cálculos de trajetória são amplamente vistos em trabalhos na área militar e espacial para lançamento de projéteis, e também em trabalhos relacionados a trajetos de aeronaves como é o caso de Bagherian e Alos (2015) e Yu, Zhou e Zhang (2019). Já o planejamento de rota, conhecido também por *Path Planning*, é definido por Coutinho, Battarra e Fliege (2018) como a determinação de uma rota, passando por determinados pontos de um espaço 2D ou 3D, desprezando a variável tempo. Estes dois conceitos apesar de serem diferentes, quando tratados juntos deixam o modelo estudado mais próximo de um cenário real.

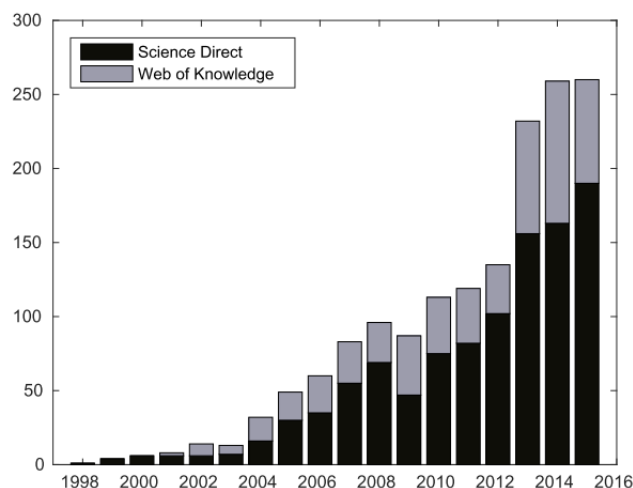
O escopo deste trabalho tem foco em apresentar uma alternativa de planejamento de rota (*Path Planning* - PP), mais precisamente rota para cobertura de área (*Coverage Path Planning*

- CPP), sendo assim, planejamento de trajetória (*Trajectory planning* - TP) não será tratado no escopo desta pesquisa.

O CPP é definido por Khan, Noreen e Habib (2017) como um problema que busca encontrar um caminho que passe por toda uma lista de pontos de GPS de uma área determinada, os chamados *waypoints*. Sendo assim, torna-se fundamental em aplicações de robótica, como: tarefas de limpeza, tarefas de inspeção e monitoramento.

Com a crescente facilidade de acesso à tecnologia dos VANTs (RAO; GOPI; MAIONE, 2016), sua aplicabilidade aumentou, e conseqüentemente aumentaram-se o número de pesquisas abordando algoritmos de planejamento de rota. São exemplos de trabalhos dentro da área de planejamento de rota: pesquisas com foco em situações críticas de resgate (QUARITSCH *et al.*, 2011), pesquisas de âmbito militar (BOLLINO; LEWIS, 2008), e outras que aplicam o planejamento de rota em ambiente urbano (FILIPPIS; GUGLIERI; QUAGLIOTTI, 2012). Esse aumento do número de trabalhos que envolvem PP pode ser observado na pesquisa realizada por Coutinho, Battarra e Fliege (2018) como mostra a Figura 8.

**Figura 8:** Número de trabalhos que abordam planejamento de rota de VANTs em fontes de informações acadêmicas, por ano de publicação



Fonte: Adaptado de Coutinho, Battarra e Fliege (2018)

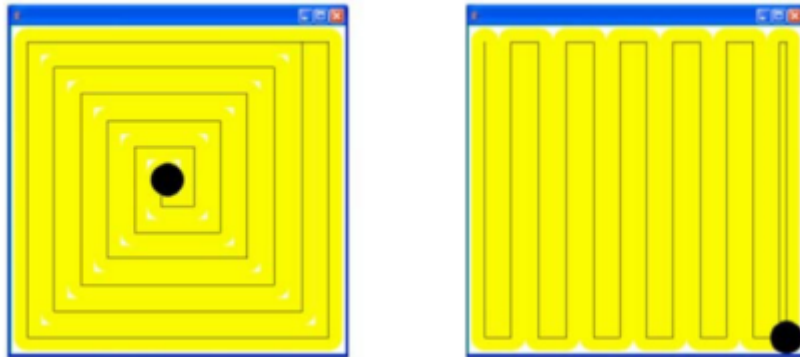
Um ponto importante citado por Khan, Noreen e Habib (2017) é a métrica da eficiência de um algoritmo CPP, que de acordo com os autores é importante considerar a taxa total de cobertura, o tempo total gasto, a distância total percorrida e a energia consumida no percurso. Já Bulusek (2015) utiliza para a medição de eficiência e análise de seus algoritmos, o total de distância percorrida e a taxa de cobertura de área. Em sua pesquisa, Bochkarev (2017) trabalhou soluções do Problema do Caixeiro Viajante Generalizado (*Generalized Travelling Salesman Problem* - GTSP) e comparou a rota obtida em seu trabalho com outras duas, anali-

sando quatro aspectos: número de instâncias do TSP, tempo de execução, total de distância da rota e porcentagem de área coberta. É por conta da variedade de métodos de avaliação e dificuldade de estabelecer um padrão de comparação entre métodos trabalhados que Bochkarev (2017) chama atenção para o desafio de medir a efetividade de um algoritmo de CPP, já que em cada aplicação os interesses são diferentes, pois um bom algoritmo de CPP para um robô que pinta paredes deve proporcionar uma parede inteiramente pintada e com uniformidade, enquanto que para um bom planejamento de rota para um VANT na aplicação de aerofotogrametria, um bom resultado seria uma rota que de maneira eficiente capturasse um conjunto de imagens que registram na totalidade a área de interesse, sendo as imagens capturadas com boa qualidade, obtidas em um intervalo de tempo constante, e mantendo a taxa de sobreposição desejada.

Os algoritmos de CPP, são considerados uma subclasse dos problemas de PP e possuem uma complexidade *NP-Hard* por se assimilarem o problema do Caixeiro Viajante (TSP) que também é *NP-Hard*. Sinha (2017) citou que por conta da complexidade deste algoritmo, nem sempre utilizar métodos simples resultam em uma solução ótima. Métodos randômicos de navegação em área, podem funcionar muito bem para algumas aplicações, como é o caso de robôs de limpeza doméstica, que cada vez que se deparam com um obstáculo, mudam aleatoriamente a direção à seguir. Já para grandes áreas, como geralmente se encontra em cenários agrícolas, este método de planejamento de rota randômico teria baixa eficiência, sendo necessário portanto, métodos mais elaborados. Por outro lado, os métodos mais elaborados possuem uma maior complexidade de modelagem e implementação. Bochkarev (2017) cita o fato de que buscando simplificar o problema de pesquisa, muitos trabalhos abordam somente a otimização de partes do problema de CPP, ou seja, acabam restringindo a solução a unicamente um ou alguns dos subproblemas do CPP como, decomposição de polígonos ou otimização da ordem de visitação dos terreno, o que faz com que a solução final não resulte em uma solução abrangente e flexível o suficiente para tratar diferentes cenários reais.

Com a diversidade de métodos existentes que podem ser utilizados nos algoritmos de CPP, fica claro a necessidade de se esclarecer os limites do modelo que está sendo proposto e as técnicas que serão utilizadas nesta pesquisa. Sinha (2017), por exemplo, esclarece para o leitor que a sua solução não trata áreas côncavas, não considera obstáculos nas área e trata o problema exclusivamente no plano 2D.

Entrando mais detalhadamente nos conceitos e técnicas dos algoritmos de CPP, Khan, Nooreen e Habib (2017) citaram em seu trabalho dois padrões de movimentos básicos que servem como base para outros padrões mais complexos de cobertura de área, são eles: *boustrophedon motion* (*back-and-forth* ou *zig-zag*) e espiral quadrático, ilustrados na Figura 9.

**Figura 9: Representação gráfica dos dois padrões básicos de preenchimento de área**(a) Padrão de cobertura espiral quadrático (b) Padrão de cobertura *boustrophedon*

**Fonte: Adaptado de Khan, Noreen e Habib (2017)**

Khan, Noreen e Habib (2017) chamam a atenção para o fato de que os algoritmos de CPP possuem três fatores importantes, que são tratados neste método de planejamento de rota proposto como subproblemas, onde cada um deles pode influenciar na eficiência geral da solução.

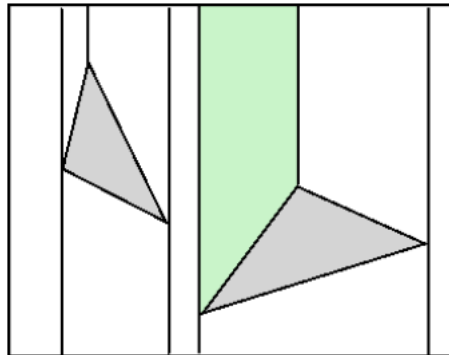
- Subproblema 1 - Decomposição de área: etapa onde são utilizadas técnicas para subdividir uma área em subáreas menores, chamadas de células, com o objetivo de diminuir a complexidade e aumentar a eficiência da solução. Sobre os algoritmos de decomposição de polígonos, Khan, Noreen e Habib (2017) citaram como dados de entrada para a resolução deste subproblema a área a ser decomposta e seus obstáculos internos, se existirem. Um diferencial do trabalho de Bulusek (2015) e de Yu e Hung (2015) por exemplo é fazer com que o algoritmo proposto pelos autores considere os obstáculos existente no cenário e planeje a rota evitando passar pelos obstáculos do cenário. Esta característica torna o modelo estudado mais preparado para uma situação real. No trabalho de Torres *et al.* (2016) cada área côncava passa por um processo de decomposição para transformá-las em múltiplas subáreas convexas. O autor diz ser importante utilizar um método de decomposição que resulte em um número pequeno de subdivisões, possibilitando assim uma otimização considerável na solução como um todo.

Principais abordagens de decomposição de polígonos:

- métodos de decomposição em células: divide a área de entrada em múltiplas subáreas menores e navega entre elas em um modelo de grafo. As subáreas podem ser percorridas utilizando algoritmos mais simples como *boustrophedon* e espiral quadrático

ilustrados na Figura 9. Exemplos: decomposição celular trapezoidal (*trapezoidal cell decomposition*) e decomposição celular de boustrophedon (*boustrophedon cell decomposition*). A Figura 10 representa uma área decomposta utilizando um método de decomposição celular exata, onde a subárea destacada representa uma célula.

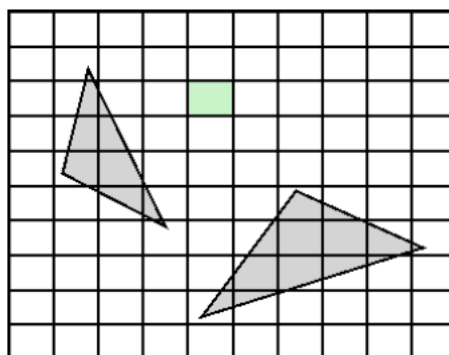
**Figura 10: Representação gráfica do método decomposição celular exata**



Fonte: Adaptado de Khan, Noreen e Habib (2017)

- métodos baseados em grade: nesta técnica o ambiente é dividido em uma grade de regiões uniformes, ou também conhecida como *grid*, o que facilita representar o ambiente e seus obstáculos. Uma das desvantagens deste método em relação ao método anterior é que o modelo do ambiente mapeado fica mais impreciso. Grandes áreas consomem muita memória para serem representadas computacionalmente. A Figura 11 representa uma área decomposta utilizando o método baseado em *grid*, onde a subárea destacada representa uma célula.

**Figura 11: Representação gráfica do método decomposição celular aproximada**



Fonte: Adaptado de Khan, Noreen e Habib (2017)

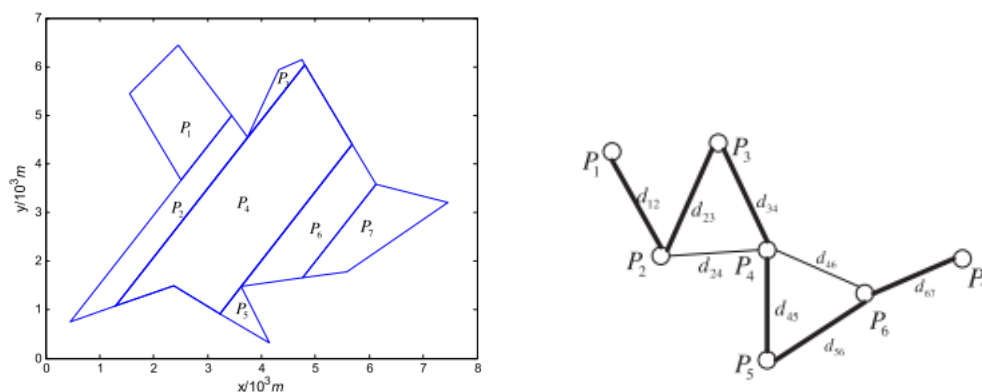
- Subproblema 2 - Determinação da direção de voo: este critério influencia no número de curvas na rota. Em trabalhos como o de Li *et al.* (2011) procurou-se diminuir o número de



curvas do trajeto pois, a curva, dependendo da configuração dos VANTs, é constituída de uma desaceleração, um giro em seu próprio eixo e uma aceleração em uma nova direção, aumentando assim o tempo para concluir percurso e o gasto de energia. Neste subproblema trabalha-se em busca de uma otimização local, focando na navegação interna das subáreas. No trabalho de Torres *et al.* (2016), após a fase de decomposição, o autor utiliza o cálculo da altura máxima de um polígono para determinar a direção que o VANT deve seguir, e posteriormente preencher o terreno com o padrão *boustrophedon* seguindo a orientação calculada.

- Subproblema 3 - Determinação da ordem de visita das subáreas: busca otimizar a ordem de visita dos terrenos, reduzindo o tempo de voo e aumentando a eficiência da solução. Após utilizado uma técnica de decomposição de área, a área decomposta representada na Figura 12(a), pode ser assimilada a um grafo, como representado na Figura 12(b), obtendo-se assim um típico caso de otimização de TSP.

**Figura 12: Comparação de uma representação gráfica e uma representação em grafo de um cenário com sete subáreas**



(a) Representação gráfica de um terreno decomposto (b) Representação em grafo de um terreno decomposto

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

Nesta fase de otimização de rota, típico problema da classe *NP-Hard*, Khan, Noreen e Habib (2017) listaram algumas técnicas que podem ser utilizadas para solucionar este tipo de problema, tais como:

- algoritmos gulosos: rápidos, foco em ótimos locais, mas nem sempre atingem o ótimo global. Exemplos de algoritmos gulosos são explicados no trabalho de *survey* de Khan, Noreen e Habib (2017): busca em profundidade, algoritmo Dijkstra, algoritmo A\*, algoritmo D\* e algoritmo Theta\*.

- algoritmos evolucionários: possuem estratégias para fugir de ótimos locais, visando o ótimo global. Exemplos de algoritmos evolucionários explicados no trabalho de *survey* de Khan, Noreen e Habib (2017): algoritmos genéticos, colônia de formigas, enxame de partículas.

Um exemplo de trabalho que busca a otimização da ordem de visitação de terrenos é a pesquisa de Sinha (2017), que utiliza algoritmos de TSP, com a intenção de descobrir uma rota de visitação mais eficiente para aplicar pesticidas em culturas utilizando VANTs. O autor desenvolveu duas versões diferentes do algoritmo de TSP, uma com abordagem gulosa de força bruta e a outra usando Dijkstra. Uma das grandes contribuições deste trabalho, são as comparações e análises realizadas entre as duas técnicas. O método apresentado pelo autor gera um arquivo de saída com os pontos de navegação da rota (*waypoints*), deste modo podem ser carregados no VANT através da estação de controle e desta forma facilmente testados em um cenário real. Ao final do trabalho o autor conclui que o modelo com Dijkstra gerou os melhores resultados.

Esta fase de otimização de ordem de visitação encontra-se também no trabalho de Torres *et al.* (2016), pois em seu método para cada subárea resultante da fase de decomposição existem quatro maneiras diferentes para aplicar o padrão de *boustrophedon* que minimiza o número de curvas do trajeto para cobertura do terreno. Sendo assim, para determinar a melhor ordem de visitação das subáreas o autor considera em seus cálculos quatro possíveis alternativas de rota para cada terreno.

Para alguns casos o ganho com a otimização da ordem de visitação de terrenos pode ser desprezível perto do ganho obtido na fase de decomposição. Como exemplo podemos observar os cenários de áreas muito próximas umas das outras, como ocorreu no trabalho de Huang (2001). Já em casos de áreas espaçadas a determinação de uma ordem de visita otimizada tende a trazer maiores ganhos que a fase de decomposição.

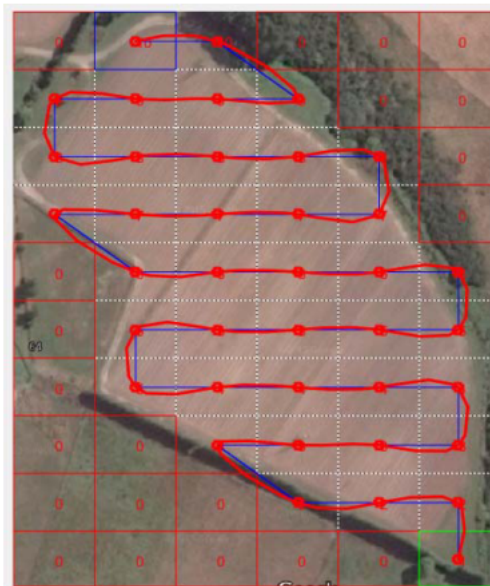
Bochkarev (2017) apresentou a etapa de decomposição de áreas em sua solução e descreveu as características de alguns métodos de decomposição celular exata, citando o problema do método de decomposição trapezoidal de criar um número alto de células, fato este, que pode aumentar a complexidade do algoritmo que determina a ordem de visitação das células, e complementa informando que a decomposição celular de *boustrophedon* pode ser considerada uma evolução da decomposição trapezoidal pelo fato de gerar um número menor de células.

Os autores Yu, Zhou e Zhang (2019) implementaram após a etapa de decomposição das áreas e antes da determinação da orientação de navegação um passo intermediário, onde as

subáreas vizinhas e de orientação de voo iguais foram reagrupadas mesmo nos casos de formatos côncavos. Desse modo em alguns casos o resultado é um cenário com menor número de subáreas, o que pode resultar em um menor número de curvas no trajeto. Um número baixo de subáreas contribui para diminuir a complexidade de se determinar a ordem de visitação entre estas subáreas.

O algoritmo proposto por Yu e Hung (2015) é muito parecido com o de Torres *et al.* (2016) e Li *et al.* (2011) por seguirem a abordagem de decomposição exata de polígonos, e contar com os três subproblemas já citados: decomposição de terreno complexos em subáreas mais simples, geração de um padrão de navegação interno em cada subárea e determinação da ordem de navegação entre as subáreas.

Após a resolução dos três subproblemas, tendo determinado o conjunto de pontos que determinará a rota, ainda existe mais um passo que influencia na eficiência da solução e é pouco abordado nos trabalhos de CPP, passo este chamado de *Path Smoothing*. O *Path Smoothing* é uma técnica utilizada para suavizar a rota obtida, considerando as restrições físicas do robô, ou VANT no caso. O VANT do tipo asa fixa por exemplo, não consegue fazer curvas bruscas, por isso considerar seu raio de curva no planejamento de rota, e suavizar a proposta de rota gera uma rota mais precisa e eficiente. São várias as técnicas de suavização existentes, desde as que utilizam métodos gráficos como curva em U, até métodos funcionais como suavização pela função de Belzier (INGERSOLL *et al.*, 2016). A Figura 13 ilustra uma rota sem suavização em azul, e a mesma rota suavizada em vermelho. Neste trabalho não será realizada a etapa de suavização, já que a solução estudada terá foco em VANTs multirotores, que além de conseguirem fazer curvas fechadas, muitos deles conseguem suavizar automaticamente a rota da missão carregada por configuração na estação de controle de solo.

**Figura 13: Representação gráfica de uma rota suavizada**

Fonte: Nam *et al.* (2016)

Ainda quanto à classificação dos algoritmos de CPP, eles podem ser divididos em dois grandes grupos: algoritmos *offline* e algoritmos *online*:

- Algoritmos *offline* – estes algoritmos utilizam dados estáticos e um conhecimento prévio do ambiente de navegação. A rota é calculada antes da missão do VANT começar. De acordo com o *survey* de Khan, Noreen e Habib (2017), a decomposição celular e os algoritmos de rede como os genéticos e colônias de formigas, são algumas das técnicas utilizadas de CPP para abordagens *offline*.
- Algoritmos *online* – Khan, Noreen e Habib (2017) apontaram que nestes algoritmos, as ações e a determinação da rota ocorrem em tempo real através de um conjunto de controladores, sensores e atuadores.

Ainda existe outra classificação para os algoritmos de CPP: eles podem ser de agentes únicos, onde um único VANT faz a cobertura total da área, ou multiagente, onde um conjunto de VANTs é utilizado para percorrer a área inteira, buscando aumentar a eficiência em áreas de grandes extensões. No trabalho de Bochkarev (2017) são apresentados estudos de algoritmos de CPP tanto para abordagens com um único quanto para mais de um VANT.

Khan, Noreen e Habib (2017) citaram um fato importante para os trabalhos de CPP em geral, apesar de existir muitas técnicas que podem ser utilizadas para planejar uma rota de

cobertura, ainda faltam na literatura métodos de análises comparativas de eficiência entre as soluções encontradas no estado da arte.

Huang (2001) chama atenção para outro fator importante nos algoritmos de CPP em geral, o tempo de execução para propor uma rota. O autor registrou em sua pesquisa que seu algoritmo levou cerca de 15 minutos para atingir a solução em seu computador de teste. Apesar do método do autor ser uma abordagem offline e necessitar de um conhecimento prévio das variáveis do ambiente, levar 15 minutos para apresentar uma rota é um fato que inviabiliza o uso desta solução em tarefas de aerofotogrametria, sendo assim, dificilmente o método de CPP proposto pelo autor seria adotado por profissionais da área de aerofotogrametria, pelo simples fato dos pilotos dependerem de janelas curtas e específicas de tempo para capturar imagens e qualquer mudança climática ocorrida no ambiente nesses 15 minutos como uma nuvem se deslocando no céu pode criar sombras nas imagens capturas e interferir na análise e resultados do processamento destas imagens.

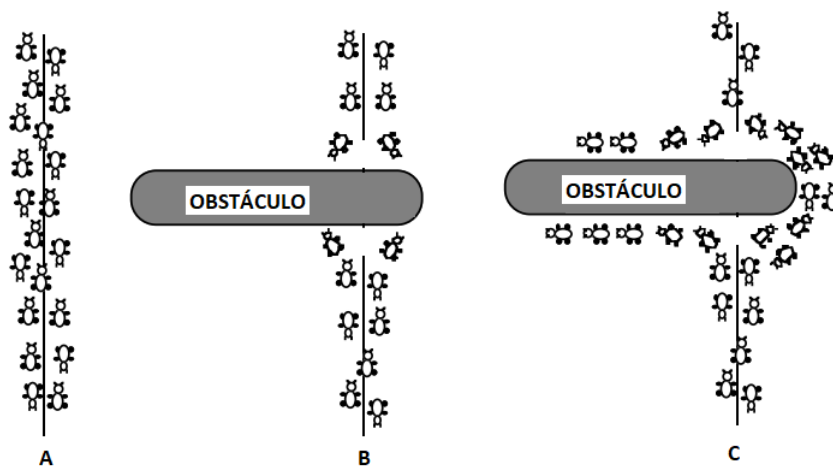
Observa-se uma tendência da maioria dos trabalhos que incluem a decomposição exata em sua solução de buscarem diminuir o número de curvas do trajeto além da distância total percorrida. Di Franco e Buttazzo (2016) realizaram um trabalho mais detalhado voltado para a modelagem de uma função de gasto de energia durante a cobertura de uma área, e obtiveram resultados que servem de suporte para outros trabalhos baseados no princípio de que diminuir o número de curvas do trajeto pode trazer ganhos de eficiência na aplicação. Outro trabalho que tem a temática parecida com a de Di Franco e Buttazzo (2016) é o de Li *et al.* (2011), que além de apresentar uma proposta de CPP utilizando decomposição celular exata, realizou estudos para provar que um trajeto com poucas curvas pode ser mais eficiente que um trajeto com mais curvas. Para isso focaram suas pesquisas na premissa de que o arrasto é maior nas curvas do que em linha reta sob as mesmas condições de velocidade e altitude.

Bochkarev (2017) também alerta sobre o excesso de número de curvas em rota, que podem consumir recursos de forma desnecessária, sendo portanto uma característica indesejável na grande maioria dos algoritmos de CPP. Outro trabalho que se baseia em diminuir o número de curvas do percurso para aumentar a eficiência da solução é o de Yu e Hung (2015), que apresentaram um método que aceita como entrada terrenos côncavos e convexos. O trabalho de Torres *et al.* (2016) também busca otimizar o CPP diminuindo o número de curvas do percurso, e se apoia nos estudos de Li *et al.* (2011), reforçando o fato de que curvas excessivas no trajeto, tendem a diminuir a eficiência do voo contribuindo com gasto de recursos energéticos em ações desnecessárias.

## 3.2 Otimização por colônia de formigas

O algoritmo de otimização por colônia de formigas, também conhecido por ACO (*Ant Colony Optimization*), foi introduzido na literatura por Marco Dorigo, e é resultado de uma adaptação de outro algoritmo chamado *Ant System* (DORIGO; MANIEZZO; COLORNI, 1991). O ACO é classificado como um algoritmo bioinspirado. Sua inspiração surgiu a partir do resultado de um experimento realizado pelo biólogo Goss *et al.* (1989), onde foi analisando o comportamento de formigas que partiam do ninho em busca de alimento. Dorigo e Gambardella (1997) explicaram em seu trabalho o comportamento que inspirou o algoritmo, utilizando o seguinte cenário: Existe uma trilha de formiga saindo do ninho até fonte de comida como retratado na Figura 14 A. Cada formiga, ao caminhar, deposita uma quantidade feromônio por onde passa. As formigas tendem a seguir os caminhos que possuem uma maior quantidade de feromônio depositado, é por este motivo, que elas seguem umas as outras. Em um certo momento um objeto é depositado no caminho, e a trilha de feromônio é interrompida, como mostra a Figura 14 B. Nesta situação, não tendo trilha de feromônio para seguir, as formigas se distribuem aleatoriamente percorrendo o contorno do objeto, continuando a depositar feromônio por onde passam, como fica retratado na Figura 14 C. Com o passar do tempo, na trilha da extremidade maior do objeto, por ser mais longa, menos formigas concluíram essa trilha, portanto recebeu menos feromônios, enquanto que na trilha mais curta, mas formiga passaram, portanto houve um maior depósito de feromônio. Apesar das duas trilhas de feromônio terem sofrido com a evaporação, a trilha curta por ter a maior quantidade de feromônio, acaba por ser mais atrativa para as formigas. Depois de algum tempo por não ter mais nenhuma formiga seguindo a trilha longa, ela acaba por evaporar totalmente, e todas as formigas passam a seguir uma única trilha.

**Figura 14: Representação gráfica do comportamento das formigas quando um obstáculo é inserido em uma trilha**



Fonte: Adaptado de Dorigo, Maniezzo e Colorni (1996)

Esse comportamento observado nas formigas retrata um tipo de inteligência coletiva, onde um indivíduo com capacidade de ação simples, não consegue alcançar uma solução complexa de um problema, mas quando em conjunto, os indivíduos estabelecem uma comunicação indireta e através de uma auto-organização são capazes de atingir soluções complexas. A auto-organização é decorrente da estigmergia. Estigmergia foi definida por Grassé e Noirot (1948) como uma forma de comunicação indireta utilizada por partes de um sistema auto-organizável que se realiza através de modificações no ambiente, ou seja, um agente produz uma alteração no ambiente, essa alteração no ambiente desperta uma ação nos outros agentes, e os estes agentes por sua vez, também sinalizam alterações no ambiente, fechando um ciclo de comunicação indireta que contribui para a auto-organização do grupo.

Dorigo percebeu que todo esses aspectos observados com base no comportamento natural das formigas de trilhar um caminho de menor distância até a fonte de alimento poderiam ser reaproveitados para solucionar problemas similares de otimização combinatória. Com base nessa associação, Dorigo buscou modelar computacionalmente o comportamento das formigas para aplicá-lo na resolução do problema do Caixeiro Viajante.

Na modelagem computacional o algoritmo de ACO é considerado uma meta-heurística bioinspirada, e possui estratégias para propor soluções ótimas ou quase-ótimas para um problema NP-Hard. Por este motivo o ACO é amplamente aplicado a resoluções de problemas combinatórios e problemas de otimização de sistemas como é o caso do Problema do Caixeiro Viajante.

Para solucionar computacionalmente o Problema do Caixeiro Viajante aplicando o algo-

ritmo ACO, o contexto do problema que se está resolvendo deve ser modelado em forma de grafo, onde cada nó deste grafo receberá uma formiga artificial. Navegando entre os nós vizinhos ao que já se encontra, cada uma das formigas deve individualmente buscar uma rota que visite todos os nós do grafo. Cada arco do grafo percorrido pela formiga recebe uma quantidade de feromônio determinada por uma taxa de deposição. O caminhar da formiga é um evento probabilístico. O tamanho do arco e a quantidade de feromônio que ele possui influencia no resultado probabilístico de qual será o próximo nó de destino da formiga. A evaporação é modelada como uma retirada da quantidade de feromônio presente no arco, de acordo com uma taxa de evaporação (GOLDBARG; GOLDBARG; LUNA, 2017). É através do controle dessa taxa de feromônio, depositando e retirando valores dos arcos, que a busca de soluções se torna mais diversificada ou mais intensa. Valores altos de feromônio intensificam a busca em um determinado trecho, mas podem fazer com que o algoritmo fique preso em mínimos locais. Já valores baixos de feromônio diversifica as buscas, mas pode atrasar a convergência do algoritmo.

Uma fator que difere do comportamento natural das formigas é que as formigas artificiais possuem memória, desta forma pode-se fazer com que ela não visite um nó já visitado anteriormente.

É através dos conceitos de *feedback* positivo, *feedback* negativo, múltiplas interações e variáveis do ambiente que buscou-se imitar computacionalmente a auto-organização das formigas (GOLDBARG; GOLDBARG; LUNA, 2017).

Atualmente existem múltiplas variações de algoritmos baseado no comportamento das formigas, o trabalho de Sivakumar e Elakia (2016) apresenta uma tabela com diversas variações, contendo seus autores e anos de publicação da técnica. O motivo de existir muitas variações é devido a busca constante de melhoria de desempenho e adaptação ao diferentes problemas. A maioria destes algoritmos de ACO contém alguns parâmetros ajustáveis em comum: o peso do feromônio na fórmula de probabilidade, o peso da distância na fórmula de probabilidade, a taxa de evaporação, a quantidade de feromônio depositada, a quantidade de feromônio inicial nas trilhas, o número de formigas da colônia e a máxima quantidade de iteração do algoritmo. As alterações realizadas nos valores destas variáveis, refletem no comportamento do algoritmo na busca por soluções.

Devido as suas características, o algoritmo ACO possui flexibilidade de adaptação à diferentes problemas, sendo aplicado em diferentes áreas. Gonsalves e Shiozaki (2015) utilizaram o algoritmo de ACO para solucionar um problema combinacional de agendamento de trens, que foi modelado como uma instância de Caixeiro Viajante Assimétrico. Yousefikhoshbakht, Didehvar e Rahmati (2013) também aplicaram o algoritmo de ACO para resolver um Problema



de Caixeiro Viajante, mas neste trabalho foram considerados múltiplos agentes. Já trabalhos como os de Cekmez, Ozsiginan e Sahingoz (2016) e Pereira (2014) utilizaram o algoritmo de ACO para realizarem um planejamento de rota de VANTs.

No trabalho de Jun-man e Yi (2012) foi chamada a atenção para um fato importante e que agregou para esta pesquisa. O autor citou que os problemas modelados como Caixeiro Viajante Generalizado (*Generalized Travelling Salesman Problem - GTSP*), que é o caso desta pesquisa, são poucos explorados na literatura. Quando acontece de existir um problema GTSP, muitas vezes sofrem adaptações e conversões para um modelo de TSP, para assim serem aplicadas técnicas mais simples e difundidas. A desvantagem é que a conversão de um modelo GTSP para TSP implica em um aumento de dimensão da matriz do problema a ser tratado, o que dependendo do cenário pode não ser uma alternativa viável. Para resolver o problema de Caixeiro Viajante Generalizado Jun-man e Yi (2012) aplicaram um algoritmo de ACO, e reforçaram a importância dos pesos do feromônio e da distância, pois estes fatores influenciam fortemente nas escolhas probabilísticas do algoritmo. Para um melhor ajuste dos valores das variáveis de controle, o autor utilizou técnicas para alterar os parâmetros de forma dinâmica durante a execução do algoritmo.

A escolha de quando utilizar o algoritmo ACO depende muito do problema que está sendo tratado, pois como todo algoritmo o ACO tem seus prós e contras. Wang, Zhang e Yan (2018) citaram em seu trabalho a desvantagem do algoritmo do ACO de facilmente ficar preso em ótimos locais, além de poder demorar para convergir. A pesquisa de Wang, Zhang e Yan (2018) teve foco na otimização do algoritmo de ACO, procurando fazer melhorias no algoritmo para evitar ficar preso em ótimos locais e ao mesmo tempo não prejudicar o tempo de convergência do algoritmo. Seus testes foram aplicados em cenários de planejamento de rota de robôs.

# Capítulo 4

## MÉTODO PROPOSTO

---

---

Este trabalho consiste em apresentar um método de planejamento de rota de cobertura otimizado aplicado a captura de imagens aéreas de uma região escolhida. A rota proposta deve minimizar o número de curvas e a distância total do trajeto, para que desta forma haja uma otimização nos gastos de recursos energéticos do VANT. O método proposto aceita como entrada cenários com múltiplas áreas, onde é aplicado um algoritmo de otimização por colônia de formiga (*Ant Colony Optimization - ACO*) para otimizar a ordem de visitação destas áreas, e buscar melhorar a eficiência da solução proposta.

As pesquisas de Li *et al.* (2011), Torres *et al.* (2016) e Di Franco e Buttazzo (2016) mostram que um maior número de curvas contribui também para um aumento do consumo de bateria. Nos estudos de Li *et al.* (2011) foi afirmado que para uma distância fixa, o tempo de voo aumenta quando o VANT realiza curvas, pois, dependendo da configuração de voo ele necessitará realizar uma parada, uma troca de direção girando em seu eixo e ainda ter que retomar sua velocidade programada em sua nova direção. Além disso, nos estudos de Bochkarev e Smith (2016) os autores chamaram atenção para vários casos onde curvas causam efeitos indesejados, e no caso do imageamento aéreo, curvas podem resultar na aquisição de imagens com angulações indesejadas. Bochkarev (2017) citou a importância de minimizar o número de curvas em trajetos de robôs, sendo essa otimização no percurso ainda mais importante para VANTs de baixa autonomia, onde cada minuto a mais de voo é importante para mapear uma área maior. Os trabalhos citados neste parágrafo seguem uma abordagem parecida e apresentaram resultados satisfatório para áreas complexas, mostrando potencial para aplicabilidade em cenários reais.

Com base nos princípios que foram observados anteriormente, o planejamento de voo proposto neste trabalho será dividido em três subproblemas: decomposição de áreas complexas em subáreas mais simples, definição de rota de navegação interna em cada uma das subáreas e definição de uma ordem de visitação otimizada das subáreas. Khan, Noreen e Habib (2017) di-

videm seu problema de CPP de forma similar e apresentaram resultados interessantes, pois desta forma puderam trabalhar cada parte da sua solução individualmente, tornando-a mais flexível e adaptável para aplicações reais.

Para o primeiro subproblema, a decomposição de áreas, é utilizado uma técnica gulosa de decomposição, já no segundo subproblema, a otimização de rota interna das subáreas, aplica-se o cálculo da altura máxima de polígonos combinado com o padrão de preenchimento *boustrophedon*. No terceiro subproblema, a otimização de ordem de navegação, o cenário é modelado como um Problema do Caixeiro Viajante Generalizado (GTSP), onde é utilizado um algoritmo ACO para obter uma ordem de visitação dos terrenos otimizadas.

Como restrições, este trabalho considera um único VANT na modelagem, e não trata obstáculos no terreno, já que no cenário da aerofotogrametria, devido a altura do voo o número de obstáculos no percurso do VANT costuma ser bem menor do que em cenários onde robôs terrestres são os agentes. Este trabalho também despreza a força do vento nos cálculos, aproveitando do fato de que os profissionais da área já evitam voos de imageamento aéreo em dias de vento forte. Para os cálculos, foi considerado o modelo do VANT como um multirotor com a capacidade de parar no ar, fazer curvas em qualquer angulação, e que consiga navegar em pontos de GPS preestabelecidos. Foi considerado para o imageamento aérea o VANT foi equipado com uma câmera Sony S100 com sistema de disparo cadencial. A solução de CPP apresentada será do tipo *offline* e o ambiente de desenvolvimento é composto por uma máquina virtual com sistema Linux (Ubuntu 16.04LTS), utilizando como linguagem de desenvolvimento Python para codificação do método de planejamento de rota, e Matlab para os estudos com os algoritmos de ACO.

Houve uma preocupação em deixar o método abstrato ao ponto de facilitar a aplicabilidade em casos reais, além de estruturá-lo de forma que pudesse facilmente gerar resultados para diferentes cenários de entrada.

Em resumo, a rota proposta é gerada conforme os seguintes passos:

- Através do Google Earth Pro, seleciona-se as áreas que precisam ser mapeadas, e exporta-se um arquivo do tipo .kml contendo as informações geográficas das áreas.
- No programa em Python configura-se as variáveis de entrada, como parâmetros intrínsecos da câmera utilizada, taxa de sobreposição de imagens e altura do voo.
- Ao executar o programa, o arquivo .kml é passado como parâmetro, e automaticamente, o programa converte as coordenadas geográficas para métricas, deixando todas as unidades de medidas em metro(m). Após esse passo realiza a decomposição dos terrenos em subáreas convexas (subproblema 1), caso necessite, e posteriormente calcula para cada

área a direção de voo que minimize o número de curvas da rota para navegação internas nas subáreas (subproblema 2).

- Após esse passo é gerado um arquivo com informações que serve de entrada para o programa em Matlab. Este arquivo funciona como um arquivo de integração entre sistemas diferentes.
- No programa em Matlab, configura-se os parâmetro do algoritmo de ACO, que ao ser executado, utiliza o arquivo com as informações anteriores para calcular uma rota otimizada para visitação das subáreas (subproblema 3). Em seguida é gerado um arquivo com a rota proposta, que pode ser carregado no *Mission Planner* para aplicações reais, caso necessário.

As sessões a seguir detalham melhor como foi abortado cada subproblema e qual técnica foi utilizada para resolvê-lo.

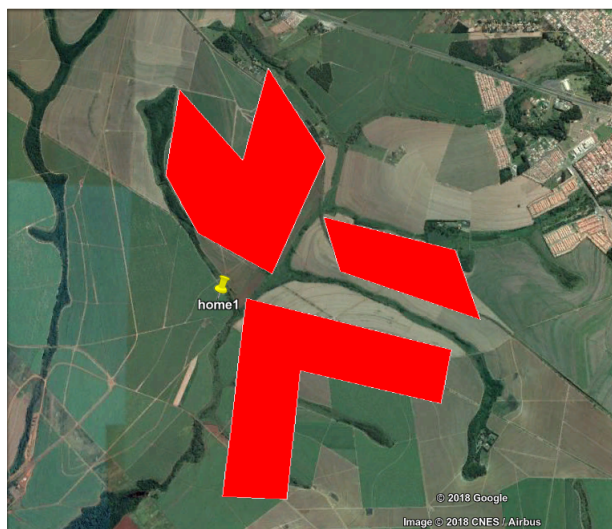
## 4.1 Entrada de dados do sistema

Como entrada do método proposto o usuário deve fornecer as informações listadas abaixo:

- Polígono da área: constituído de uma lista de coordenadas GPS que representam o perímetro da área a ser sobrevoada. Estes pontos de latitude e longitude podem ser capturados através da ferramenta *Google Earth Pro*, como retratado na Figura 15 pelos polígonos em vermelho, que posteriormente são exportados em um arquivo de terminação .kml;
- Ponto de lançamento: representa a coordenada GPS de onde o piloto decolará o VANT, ou seja, deve ser um local acessível. Para este trabalho considera-se que o ponto de lançamento também será o mesmo ponto de retorno do VANT. Este ponto também deve estar presente no arquivo .kml gerado no *Google Earth Pro*.
- Dimensões do sensor: é necessário informar ao algoritmo o tamanho focal (mm), altura (mm) e largura (mm) do sensor utilizado para capturar as imagens;
- Altura de voo: esta variável está estritamente relacionada com a resolução desejada nas imagens. Para definir a altura de voo é necessário saber qual o valor de GSD desejado, e quais as dimensões do sensor da câmera que será utilizado na captura das imagens. Portanto a altura do voo deve ser calculada previamente e informada ao algoritmo. GSD (Ground Sample Distance), como já explicado na seção 2.2, é o valor de quantos centímetros do solo será representado em um pixel da imagem;

- Porcentagem de sobreposição de imagens: o fato das fotos terem sobreposições, ou seja, cada área do terreno deve aparecer em mais de uma foto, torna possível a junção das imagens e a criação dos ortomosaicos, portanto deve ser informado ao sistema um valor de porcentagem que será atrelado as sobreposições laterais e verticais utilizadas no planejamento de rota de cobertura;

**Figura 15: Representação gráfica de uma cenário com terrenos seleccionados no *Google Earth* para serem processadas**

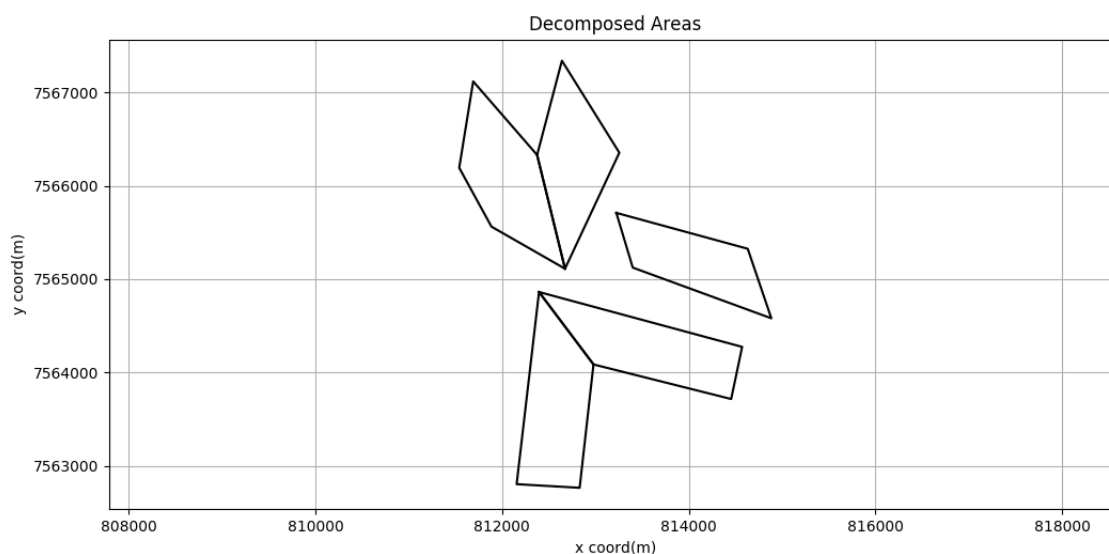


**Fonte: Própria (2019)**

Configurado os parâmetros de câmera e de voo, o programa em Python recebe o arquivo .kml como parâmetro e começa o processamento das informações, buscando tratar primeiramente terrenos côncavos, neste trabalho mapeado como o subproblema de número 1.

## 4.2 Subproblema 1 – Decomposição de polígonos

A fase de decomposição tem como objetivo dividir uma área complexa, representada por um polígono não regular em subáreas mais simples, buscando sempre transformar polígonos de formato côncavo em polígonos convexos, onde possa ser aplicado o padrão de *boustrophedon* com maior facilidade. A Figura 16 mostra que das três áreas iniciais, duas eram côncavas e por esse motivo passaram pelo processo de decomposição resultando em subáreas convexas, totalizando cinco polígonos.

**Figura 16: Cenário resultante do processo de decomposição das áreas de entrada**

**Fonte: Própria (2019)**

Sendo assim os polígonos de entrada que já estiverem no formato convexo não passarão por decomposições. Já nos polígonos côncavos serão aplicados uma técnica de decomposição gulosa para transformá-los em múltiplos polígonos convexos.

Polígonos côncavos possuem um ou mais vértices reflexos. Este tipo de vértice se caracteriza por ter uma angulação interna maior que 180 graus. Desta forma, decompor o polígono concavo em convexo se resume a dividi-los de forma à eliminar todos os vértices reflexos. Dependendo do polígono, esta tarefa pode ter a complexidade de um problema *NP-Hard*, como é o caso quando o polígono possui regiões internas que devem ser evitadas (KEIL; SNOEYINK, 2002).

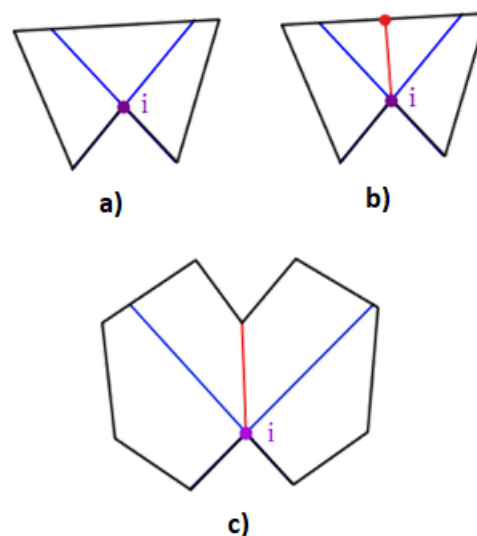
Existem áreas de estudos e diversos trabalhos com foco unicamente em como decompor polígonos visando uma solução ótima, que gere o menor número de polígonos no menor tempo possível. Neste trabalho, para decompor os polígonos, foi utilizado uma biblioteca em Python denominada *poly\_decomp*. Esta biblioteca possui dois métodos de decomposição. O primeiro método utiliza força bruta, analisando múltiplas possibilidades de remover os vértices reflexos e escolhe a opção que resulta na menor diagonal. Apesar de encontrar o resultado ótimo, este método leva tempo para apresentar a solução. O segundo método da biblioteca, tem foco na agilidade. Sendo assim, este método é de mais rápida execução que o primeiro, e apresenta uma solução em uma quantidade menor de tempo, mas nem sempre apresentará um resultado ótimo.

Pensando na aplicabilidade do algoritmo, a apresentação da solução em um curto espaço de

tempo se torna indispensável, já que muitas vezes o profissional da aerofotogrametria depende de uma curta janela de tempo para planejar e realizar o voo antes que o ambiente mude, que sombras se desloquem, atrapalhando a captura das imagens e posteriormente a análise imagens. Por conta deste motivo, a segundo método da biblioteca foi escolhido para ser aplicado no algoritmo.

Este método de decomposição funciona da seguinte maneira: o algoritmo parte de um vértice aleatório do polígono e itera sobre todos os outros até que encontre um vértice reflexo, representado na Figura 17 pela letra *i*. Ao encontra-lo, as arestas que o constituem são prolongadas até que encontrem alguma outra aresta, formando assim uma área interna, exemplificado na Figura 17(a). Caso não exista nenhum outro vértice dentro desta área, será criado um novo vértice no centro da aresta oposta, e o vértice reflexo será conectado a ele, como ilustra a Figura 17(b). Já no caso de existir mais de um vértice dentro da abertura do ângulo, será escolhido o vértice que resultar na menor distancia de ligação, Figura 17(c).

**Figura 17: Representações gráficas do comportamento do algoritmo de decomposição**



**Fonte: Disponível em: <https://mpen.ca/406/bayazit> Acessado em: agosto de 2018**

É importante criar o menor número possível de subáreas nesta fase, pois, assim como citado por Torres *et al.* (2016), poucas áreas refletem em um menor número de possibilidades de rotas para a otimização da ordem de visitação, ou seja, restringe o espaço de busca das soluções.

Após a fase de decomposição, o algoritmo segue para a fase de determinação de orientação de voo.

### 4.3 Subproblema 2 – Navegação interna das subáreas

O objetivo desta fase é calcular uma rota interna em cada um dos polígonos resultantes da fase anterior, buscando a orientação de voo que gere o menor número de curvas dentro de cada subárea. Para isto foi utilizado a técnica do cálculo de altura máxima do polígono, abordada nos trabalhos de Li *et al.* (2011) e Torres *et al.* (2016). Este cálculo consiste em medir a distância existente entre cada vértices e aresta do polígono, para descobrir entre qual vértice e aresta se encontra a maior distância, assim como representado no Algoritmo 1 e na Figura 18.

---

**Algoritmo 1:** Calcula a maior altura de um polígono.

*distancia(a, v)*: Distância Euclidiana entre aresta *a* e o vértice *v*

---

**Entrada:** lista de vértices do polígono, lista de arestas do polígono

**Saída:** Pontos que geraram a maior distância no polígono

**início**

**repita**

    max\_dist = 0;

**repita**

**se** *distancia(a, v) > max\_dist* **então**

        max\_dist = *distancia(a, v)*;

        vertice\_referencia = *v*;

**até** *percorrer todos os vértices*;

**se** *max\_dist < optimal\_dist* **ou** *é a primeira aresta* **então**

      optimal\_dist = max\_dist;

      aresta\_referencia = *a*;

**até** *percorrer todas as arestas*;

**retorna** *vertice\_referencia, aresta\_referencia*;

**fim**

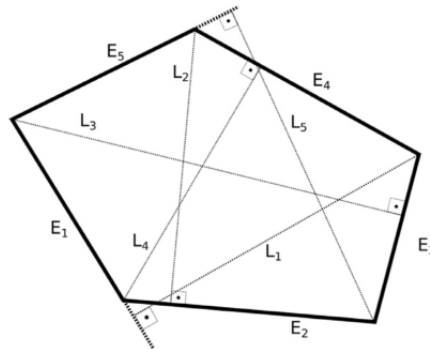
---

**Fonte:** Adaptado de Torres *et al.* (2016)

Seguindo a orientação do segmento de reta de maior distância, será aplicado o padrão *boustrophedon*. Aplicado desta forma, a padrão resulta em um menor número de curvas, comparado a outras possíveis angulações. O mesmo processo se repete em cada um dos polígonos existentes, desta forma cada um deles possuirá sua própria orientação de voo.



**Figura 18: Representação gráfica das comparações de distância realizadas entre cada vértice e aresta do polígono que acontecem no algoritmo de cálculo de direção de voo.**



Fonte: Adaptado de Torres *et al.* (2016)

Para aplicar o padrão de *boustrophedon* é necessário calcular a distância entre linhas, que dependerá do valor de sobreposição escolhido. A seguir é apresentado um exemplo de cálculo de distância entre as linhas de voo, os parâmetros que devem ser considerados são:

- Altura do voo ( $d$ ): deve ser calculada previamente, depende da aplicação.
- Porcentagem de sobreposição lateral ( $ohp$ ): valor que indica a porcentagem de sobreposição entre uma imagem e outra na direção lateral.
- Sobreposição lateral em metros ( $oh$ ): valor em metros de sobreposição entre uma imagem e outra na direção lateral.
- Distância focal ( $fl$ ): distância focal da câmera, sendo esta uma característica intrínseca do sensor.
- Largura do *footprint* ( $fpw$ ): valor que indica qual a distância em metros a câmera conseguirá capturar do solo no sentido horizontal.
- Largura do sensor ( $sw$ ): valor em milímetros da largura do sensor.

Cálculo da distância entre as linhas de voo:

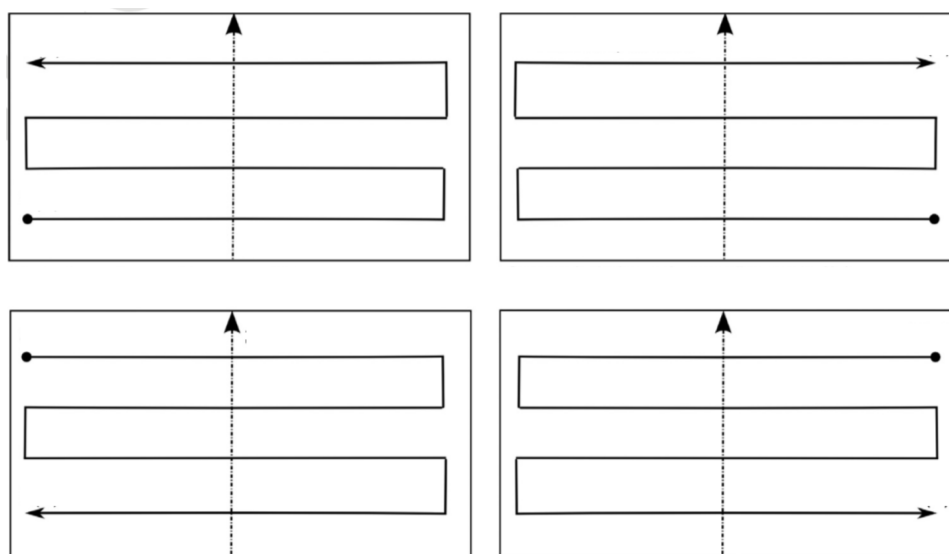
$$fpw = (d/fl) * sw \quad (4.1)$$

$$oh = fpw * (1 - ohp) \quad (4.2)$$

Portanto a distância entre linhas deve corresponder ao valor de sobreposição em metros encontrado, e o padrão de *boustrophedon* deve ser traçado seguindo este valor de espaçamento.

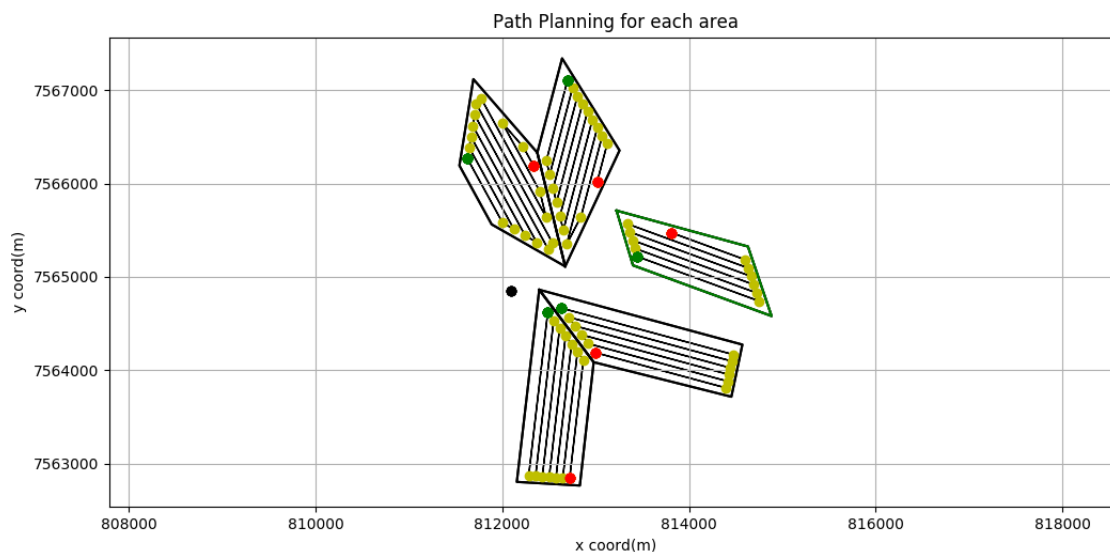
Uma vez aplicado o movimento de *boustrophedon* para a orientação de voo calculada, existem quatro possibilidades de rotas diferentes que resultam no mesmo número de curvas, como ilustra a Figura 19. As quatro possibilidades devem ser levadas em consideração na fase de otimização de ordem de visitas (subproblema 3), pois apesar de possuírem o mesmo número de curvas, o ponto de início e fim da rota influenciará na distância total final da solução. São estas quatro possibilidades de rota em cada polígono que fazem com que o problema de otimização seja do tipo GTSP, e considerando que em cada terreno somente uma rota pode ser escolhida, o problema pode ser especificado como E-GTSP (*Equality Generalized Travelling Salesman Problem*).

**Figura 19: Quatro alternativas de rotas diferentes que minimizam o número de curvas do trajeto para um dado polígono**



**Fonte: Própria (2019)**

A Figura 20 ilustra o cálculo da orientação de voo para cada área representada na Figura 16, onde somente uma das quatro possíveis rotas está representada em cada um dos polígonos. Os pontos em verde representam pontos de entrada no terreno, os pontos em vermelhos são pontos de saída, enquanto os em amarelo representam coordenadas intermediárias do trajeto.

**Figura 20: Representa o gr fica de rotas internas resultantes do c lculo da orienta o de voo**

**Fonte: Pr pria (2019)**

Para a determina o da ordem de visita o das  reas, juntamente com a escolha de qual alternativa de rota ser  utilizada em cada uma,   utilizado um algoritmo de otimiza o por col nia de formigas (ACO). Este tipo de algoritmo foi inicialmente aplicado justamente para a resolu o de Problemas de Caixeiro Viajante, e sua flexibilidade permite a adapta o para a resolu o do problema de GTSP deste estudo.

#### **4.4 Subproblema 3 – Otimiza o da ordem de visita o das sub reas**

Ap s decomposto cada pol gono em sub reas (subproblema 1) e determinado a dire o de voo que minimize o n mero de curvas da rota (subproblema 2), neste passo ser  tratado a otimiza o da ordem de visita o entre as diferentes sub reas, sendo que em cada uma existem quatro alternativas de rotas existentes que devem ser consideradas.

Para tratar este problema, classificado como do tipo GTSP, foi implementado no MATLAB um algoritmo que otimiza a ordem de visita o utilizando a t cnica de ACO (*Ant Colony Optimization*). Foi criado um arquivo de integra o entre o algoritmo em Python e o em MATLAB. Ao final do c lculo das orienta o de voo (subproblema 2), o algoritmo em Python salva no arquivo de integra o informa o que s o utilizadas como entrada para o algoritmo ACO. Atrav s desta forma de troca de informa o entre os programas foi poss vel unificar a

execução e deixar a troca de informação entre os programas transparente para o usuário.

Na pesquisa de Colorni, Dorigo e Maniezzo (1992) foi sugerido três tipos de implementações de algoritmos ACO: ANT-density, ANT-quantity e ANT-cycle. Neste trabalho, a implementação foi baseado no tipo ANT-cycle. A diferença entre este algoritmo e os outros, é que o feromônio é depositado na trilha por todas as formigas após cada uma delas atingirem uma solução. Baseado neste modelo, o fluxo implementado pode ser verificado Algoritmo 2.

---

**Algoritmo 2:** Fluxo macro do algoritmo ACO implementado

---

**início**

**repita**

    Inicialização\_das\_formigas()

    Movimentação\_das\_formigas()

    Atualização\_de\_feromônios()

    Atualização\_da\_melhor\_rota()

**até** atingir o máximo de iterações configurada;

**fim**

---

**Fonte: Própria (2019)**

As distâncias existentes entre cada ponto de saída e entrada calculados no subproblema 2 foram convertidos de forma automatizada e representados em forma de uma matriz de adjacência. A matriz criada foi passada para o algoritmo de ACO através do arquivo de integração, onde foi interpretada como um grafo.

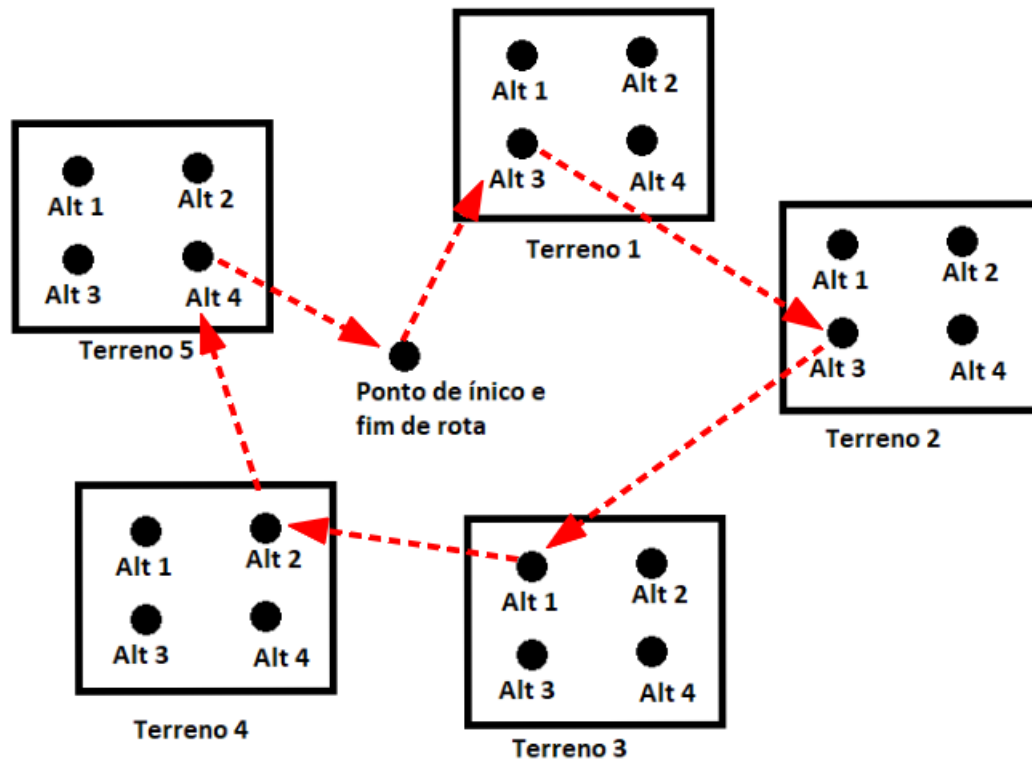
Por se tratar de um problema onde o agente é o VANT, cada um dos nós do grafo passam a ter múltiplos vizinhos, já que o VANT possui a capacidade de sair de qualquer ponto de saída de um terreno e ir em direção a qualquer ponto de entrada. Desta forma, praticamente toda rota existente é uma alternativa de caminho possível que deve ser analisada.

As únicas rotas que foram restritas na matriz de adjacência foram as que interligam alguma das quatro alternativas de rota a outra alternativa de mesmo terreno. Foi criada esta restrição para evitar que o agente possa navegar entre alternativas de mesmo terreno, já que elas não coexistem em uma mesma solução. Somente uma alternativa de cada terreno deve ser escolhida em cada proposta de rota otimizada.

A Figura 21 representa um cenário real representado como grafo, onde existem cinco terrenos, e cada um deles possui quatro alternativas de rotas internas. Cada uma das quatro alternativas de navegação interna de um terreno, como ilustrado anteriormente na Figura 19, difere

umas das outras quanto à: coordenada geográfica de entrada no terreno, coordenada geográfica de saída no terreno e comprimento da rota.

**Figura 21: Representação gráfica do modelo GTSP onde a rota escolhida deve partir e retornar para o mesmo nó, visitando somente um nó de cada terreno**



Fonte: Própria (2019)

Na implementação do algoritmo de ACO, as formigas artificiais representam o VANT no papel de agente do sistema.

As formiga artificiais foram implementadas com a característica de possuírem memória, ou seja, elas conseguem saber se já passaram anteriormente por um mesmo caminho. Esta memória é representado por uma lista contendo a identificação do nó do grafo que já foi visitado por ela. Na literatura esta técnica também é conhecida por lista Tabu, primeiramente tratada com detalhes no trabalho de Glover, Laguna e Marti (2007).

Como listado anteriormente, o primeiro passo do algoritmo ACO é a inicialização das formigas. Nesta fase cada uma das N formigas são distribuídas aleatoriamente em cada um dos N nós do grafo. O número de formigas é uma variável ajustável do algoritmo, sendo que neste caso foi utilizado uma quantidade de formigas igual ao número de nós do grafo. Na inicialização do

algoritmo a memória das formigas é apagada, deixando somente a informação do nó de partida de cada uma.

Na fase de movimentação das formigas, acontecem uma série de iterações. Em cada iteração as formigas são forçadas saírem do nó que estão e se movimentarem para um outro nó ainda não visitado. Esta escolha é realizada através de uma função probabilística, que considera a distâncias dos nós vizinhos e a quantidade de feromônios existentes nas trilhas, para assim determinar dentre o nós que ainda não foram visitados, qual tem a maior probabilidade de ser o próximo escolhido.

O algoritmo possui variáveis de controle para ajustar o quanto as variáveis taxa de feromônio e comprimento da trilha influenciam no cálculo probabilístico do próximo nó a ser visitado.

Foi implementado também um fator randômico nas formigas. Se através de um sorteio for escolhido que não se deve fazer cálculo probabilístico para determinar o próximo destino, a formiga, por sua vez, deverá escolher seu destino de maneira aleatória, sorteando um nó dentre os que ainda não foram visitados. O peso deste fator randômico no algoritmo pode ser alterado e até mesmo desligado através de uma variável de ajuste.

Esta fase de escolha e movimentação das formigas se repetirá até que elas tenham atingindo uma solução, ou seja, até que tenham visitado todos os nós do grafo.

Ao fim desta fase acontece o passo da atualização dos feromônios das trilhas. Nesta implementação do algoritmo ACO, é depositado uma quantia  $Q$  de feromônio em todas as trilhas que as formigas caminharam, onde essa quantidade é multiplicada pelo número de formigas que passaram pela trilha. A quantia  $Q$  de feromônio é definida por uma variável de ajuste no algoritmo. Neste mesmo passo acontece também a evaporação do feromônio, ou seja, é retirado uma quantidade de feromônio, também definida em variável de ajuste, de todas as trilhas do grafo. Após este passo, são comparados todas a soluções de cada formiga, onde o caminho de menor distância será eleito a melhor solução até o momento, e com isso o ciclo se repete, iniciando as formigas novamente e realizando mais uma iteração, buscando um caminho menor do que o já conhecido. Este ciclo se repetirá até o número estipulado de vezes, ajustado em variável de controle do algoritmo. Após a última iteração, o menor caminho calculado será proposto pelo algoritmo como o caminho otimizado, que consiste em pontos de GPS que o VANT deve sobrevoar para realizar a cobertura de todas as áreas de maneira otimizada.

Um dos maiores desafios de se utilizar o algoritmo de ACO é parametrizá-lo corretamente para que atinga uma solução otimizada em um curto espaço de tempo. A definição dos valores

das variáveis de controle do algoritmo utilizados neste trabalho, foi realizada através de um teste de força bruta, que com base em valores observados em outros outros trabalhos como o de Dorigo, Maniezzo e Colorni (1991), montou-se uma tabela com todas as possíveis combinações diferentes para os valores de parâmetros, sendo eles: peso do feromônio, peso da distância, taxa de evaporação e quantidade de feromônio depositada, conforme a Tabela ?? onde para cada combinação foram repetidas 10 execuções do algoritmo de ACO em um cenário de teste, cenário este ilustrado na Figura 22, sendo observado o tempo de convergência da solução e a rota mais curta obtida, para selecionar as combinações de melhores resultados.

**Tabela 2: Tabela com os valores e variáveis utilizados para gerar os lista de 625 combinações de parâmetros de controle do algoritmo ACO**

	Valor 1	Valor 2	Valor 3	Valor 4	Valor 5
Parâmetro A - Peso do feromônio	0.0	0.5	1	5	10
Parâmetro B - Peso da distância	0	1	5	10	20
Parâmetro C - Taxa de evaporação	0.3	0.5	0.7	0.9	0.999
Parâmetro D - Quantidade de feromônio depositada	1	100	1000	5000	10000

**Fonte: Própria (2019)**

Para facilitar este processo de parametrização do algoritmo ACO, foi implementado uma lógica que mantém um histórico de execuções do programa, registrando qual cenário está sendo tratado, as variáveis de entradas, os valores das variáveis de controle, o tempo gasto de execução, e também a própria solução obtida.

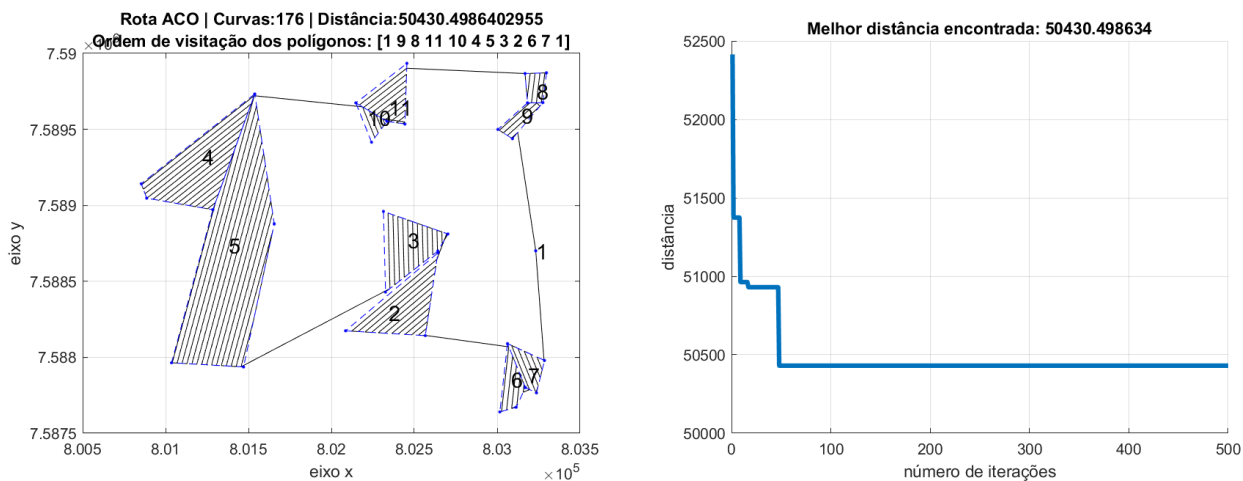
**Figura 22: Cenário de teste para determinação dos parâmetros de controle do algoritmo de ACO**

**Fonte: Própria (2019)**

Dentre as 625 opções de combinações existentes, algumas delas apresentaram resultados bons o suficientes de serem utilizados no planejamento de rotas, por apresentarem resultados quase-ótimos em um tempo aceitável para uma aplicação real de aerofotogrametria. Dentre as melhores configurações encontradas, para os testes de ACO deste trabalho, a combinação de parâmetros foi a seguinte: peso do feromônio = 0.5, peso da distância = 20, taxa de evaporação = 0.7 e quantidade de feromônio depositada = 1. A Figura 23(a) representa o algoritmo ACO resolvendo o cenário de testes com os parâmetros indicados. O limite de número de iterações estabelecido foi 500, entretanto nas 10 execuções do algoritmo não foi preciso mais que 100 iterações para atingir um resultado bom o suficiente para a aplicação, neste caso a distância mínima da rota encontrada foi de 50430,49 metros. Este resultados podem ser observados na Figura 23(b).



**Figura 23: Resultado do algoritmo ACO para o cenário de teste com utilizando os parâmetros ajustados**



(a) Representação gráfica do resultado do ACO no cenário de teste (b) Gráfico de convergência do ACO no cenário de teste

Fonte: Própria (2019)

## 4.5 Proposta de validação e de análise de resultados

Como já citado anteriormente, este trabalho baseia-se em princípios estudados por Li *et al.* (2011) e Di Franco e Buttazzo (2016), portanto considera-se que diminuir o número de curvas e a distância total de voo contribui positivamente para a economia de bateria. Sendo assim estas duas variáveis, após calculadas, serão fatores de comparação com outras possíveis soluções, no caso, o *Mission Planner*.

Nesta fase de validação dos resultados, este trabalho pretende contribuir trazendo uma visão de análise que faça a junção de conceitos acadêmicos com experiências de campo que remetem ao dia a dia dos profissionais da área, deixando claro quais tipos de cenário o método apresentou resultado bons e quais cenários o método não foi adequado. A maioria dos trabalhos encontrados na revisão sistemática validam a solução proposta utilizando terrenos criados artificialmente ou apresentando somente um único estudo de caso real. Este fato, quando analisado pelo ponto de vista de profissionais da aerofotogrametria não se trata de uma estratégia adequada para validação de um método de planejamento de rota, já que os pilotos lidam em seu dia a dia com diferentes formas de terrenos e um algoritmo que atinja um bom resultado para um caso pode ser ruim para outro. Por conta do interesse de integrar a visão do lado prático da aerofotogrametria nas análises dos resultados, as comparações serão realizadas no *Mission Planner*, que é um *software* de planejamento de voo utilizado amplamente no mercado profissional.

Para processar no *Mission Planner* o mesmo cenário anteriormente analisado no método proposto, será utilizado o arquivo .kml extraído do *Google Earth Pro*, que contém as coordenadas geográficas das áreas de interesse. O arquivo será carregado no *Mission Planner* através da opção *Map Tool / KML Overlay*. As opções citadas se localizam dentro do menu de planejamento de voo do *Mission Planner*. Após este procedimento, as áreas de interesse aparecerão desenhadas no mapa do programa, permitindo assim que uma após a outra, seja realizado o seguinte processo: delimitação do perímetro do terreno, preenchimento da área através da opção *Auto WP / Survey Grid*, que criará a cobertura da área com base nos valores das variáveis de voo fornecidos, como parâmetros intrínsecos da câmera utilizada, taxa de sobreposição de imagens e altitude do voo. Após repetir o processo com todas as áreas carregadas, resultará em uma única rota que passa por todos os terrenos do cenário.

Chama-se atenção para o fato de que o algoritmo de planejamento de rota do *Mission Planner* atualmente não computa a melhor ordem de visita quanto o cenário possui múltiplos terrenos, o algoritmo se limita a apresentar uma rota única, passando pelos terrenos na mesma ordem em que foram processados.

O resultado obtido do planejamento de rota *Mission Planner* será confrontada com a rota gerada no método proposto após resolver os subproblemas 1, 2 e 3. Estas rotas resultantes são compostas por uma lista de pontos de GPS, também conhecido na literatura por *waypoints*, que representa os locais por onde o VANT deverá percorrer enquanto captura imagens do percurso. Para a comparação entre os métodos e análise de resultados, será considerado duas variáveis: número de curvas e distância total do percurso.

# Capítulo 5

## RESULTADOS E ANÁLISES

---

---

Neste capítulo são apresentados diferentes cenários de testes e suas respectivas análises. Cada um dos cenários apresentados representa um conjunto de terrenos que necessita ser mapeado através de imagens aéreas e para realizar esta tarefa precisa ser determinado um plano de voo. O cenário pode ser composto por um ou mais terrenos agrícolas reais. Para efeito de comparação, em cada cenário serão aplicados dois algoritmos diferentes de planejamento de rota de cobertura. Após a execução dos algoritmos, resultarão em duas propostas diferentes de rota para serem utilizadas no processo de aerofotogrametria. Apesar das duas rotas propostas resultarem em um arquivo que pode ser diretamente carregado no VANT para um voo real, os dois métodos serão comparados estatisticamente considerando duas variáveis: número de curvas e distância total do percurso. Os testes possuem cenários de diferentes complexidades, e serão apresentados partindo dos mais simples para os mais complexos. Foram criados quatro grupos de testes, sendo eles:

- Cenários com um único terreno de formato convexo
- Cenários com um único terreno de formato concavo
- Cenários com cinco terrenos
- Cenários com dez terrenos

Cada grupo de teste é composto por cinco cenários diferentes de características similares. Para efeito de comparação foram utilizados os mesmo dados de entrada em cada um dos métodos aplicados, sendo eles:

- Altura do voo: 100 metros

- Taxa de sobreposição das imagens horizontalmente e verticalmente: 80%
- Dimensões do sensor: tamanho focal de 5.2mm , largura do sensor de 7.6 mm e altura do sensor de 5.7mm
- Ponto de lançamento do VANT e delimitação dos terrenos de interesse: representados por pontos GPS, contendo latitude e longitude, exportado do *Google Earth Pro* em um arquivo de formato .kml.

Para cada um dos cenários serão apresentadas imagens dos terrenos capturados no *Google Earth Pro*, juntamente com as imagens das rotas obtidas no *Mission Planner* e no método proposto. Em todos os gráficos as unidades de medidas apresentadas estão em metro(m). O conjunto de imagens será seguido de uma tabela comparativa contendo os valores das duas variáveis de interesse, a diferença percentual entre elas e a ordem em que foram visitados os terrenos.

Para todos os cenários de testes o ponto de lançamento do VANT será o mesmo ponto de retorno. Quando apresentadas as sequências de visitação dos terrenos, o ponto inicial sempre aparecerá incluso, sendo ele representado como o terreno de número 1. Toda proposta de rota deve começar e terminar no ponto de lançamento, assim como na aplicação real.

## 5.1 Cenários de terreno único: formato convexo

### 5.1.1 Cenário 1

Como primeiro cenário de teste, foi escolhido através da ferramenta do *Google Earth Pro*, o terreno de formato convexo delimitado na Figura 24(a) pela cor vermelha. O ponto marcado em amarelo representa o local de lançamento e retorno do VANT. Em aplicações reais este local costuma ser uma área de fácil acesso perto do terreno que deseja ser mapeado. Em se tratando de VANT multirrotor, por conta de sua decolagem vertical não é necessário um grande espaço para se iniciar o voo. O terreno representado na Figura 24(a) foi carregado no sistema através do arquivo .kml e interpretado pelo algoritmo como o polígono representado na Figura 24(b). Com base nos polígonos contidos no arquivo .kml, os dois métodos foram executados, utilizando os mesmos parâmetros de configuração.

A Figura 24(c) representa a rota proposta pelo *Mission Planner* quando utilizado a opção de planejamento de rota *Survey*. Os pontos em verde representam cada coordenada GPS da

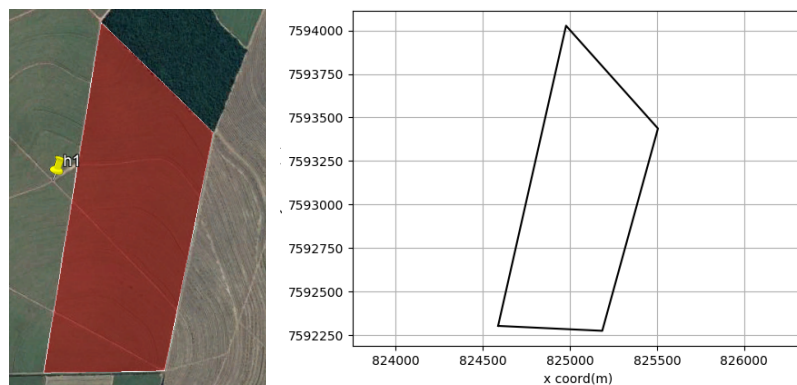
rota, tratada dentro do *Mission Planner* como *waypoints*. Os seguimentos de linha em amarelo representam a rota de voo que o VANT deverá seguir.

No *Matlab*, através dos pontos da rota proposta pelo *Mission Planner* é possível identificar a distância total e a quantidade de curvas do percurso. A representação gráfica da rota do *Mission Planner* é retratada na Figura 24(d). As unidades dos eixos  $x$  e  $y$  dos gráficos se encontram em metros ( $m$ ), e correspondem à valores resultantes de um processo de conversão de coordenadas geográficas (latitude e longitude) para coordenadas métricas.

Já no método proposto, antes de computar as variáveis, é necessário rodar o algoritmo ACO utilizando os dados provenientes das fases anteriores (decomposição e orientação de voo), para que quando executado o algoritmo de ACO, dentre as quatro possíveis alternativas de rota do terreno, possa ser escolhido uma para compor um caminho otimizado.

É de conhecimento que para casos como este, onde existe somente um terreno a ser mapeado, não existe necessidade de aplicação do algoritmo ACO, pois como são somente quatro alternativas de rota a serem consideradas, a obtenção do resultado ótimo é de simples obtenção por força bruta. Apesar disto, pode ser observado no gráfico de convergência ilustrado na Figura 24(e) que a obtenção do resultado pelo ACO também foi instantânea. Após a execução do algoritmo ACO, a rota resultante passa pela mesma análise de distância total e número de curvas que a rota do *Mission Planner*, representada na Figura 24 (f) para que desta forma possa ser criado valores para uma comparação.

**Figura 24:** Fases do processo de comparação de rota entre *Mission Planner* e método proposto para o cenário 1

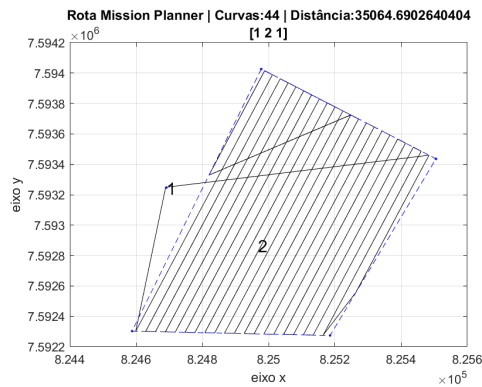


(a) Terreno selecionado no Google Earth Pro

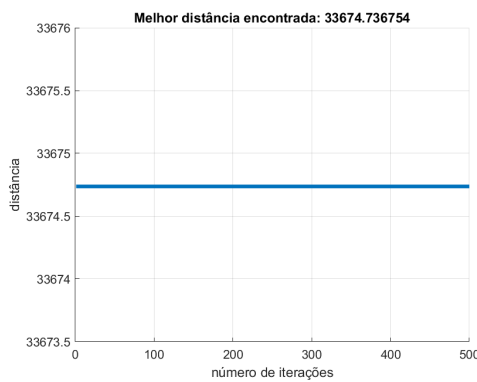
(b) Terreno representado no algoritmo



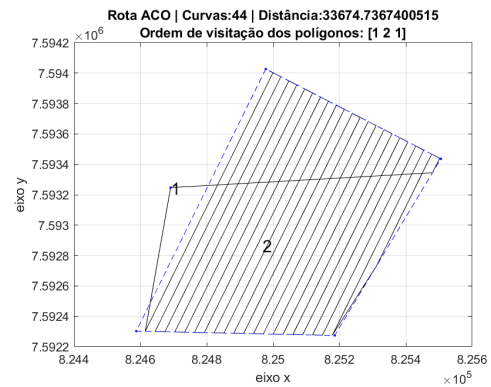
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

A Tabela 3 contém o número de curvas e a distância total do percurso dos planos de voo proposto por cada um dos métodos. Adicionalmente, para uma melhor análise, é apresentado um indicador de variação percentual do método proposto em relação ao método do *Mission Planner*.

**Tabela 3: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 1**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	44	35064.69	2	2	1-2-1
Rota 2 - Método proposto	44	33674.73	2	2	1-2-1
Variação percentual	0	-3.96			

Fonte: Própria (2019)

O resultado deste primeiro cenário retratado na Tabela 3, indica que para esta área de interesse de formato simples, os resultados dos dois métodos foram praticamente os mesmos. Este fato de apresentar resultados semelhantes é interessante principalmente por servir de validador da implementação realizada dos algoritmos, já que em uma área convexa simples não eram esperados grandes diferenças entre as rotas propostas.

Percebe-se que na rota 2 da Tabela 3, pela área de interesse já se tratar de uma forma convexa, o algoritmo não realizou decomposição do polígono, sendo assim, o número de terrenos tratado no algoritmo foi o mesmo do início ao fim do processamento.

Através da Tabela 3 nota-se que a ordem de visitação dos terrenos foi a mesma nos dois métodos, partindo do local de lançamento, considerado como terreno 1, visitando o terreno 2 e retornando para o terreno 1. Pode-se observar também que com base nos valores de variação percentual, a rota do método proposto conseguiu uma economia de 3.96% de distância total em relação ao método do *Mission Planner*.

Os dois métodos possuem o movimento de *boustrophedon* como base para preenchimento das áreas, mas diferente do método proposto, a rota no *Mission Planner* pode interromper o preenchimento no padrão *boustrophedon* para realizar outros tipos de movimento caso julgue ser necessário para preencher completamente uma área.

Outro fator que difere entre os métodos está relacionado com a orientação de preenchimento. Enquanto no método proposto a orientação de aplicação do padrão *boustrophedon* sempre segue a maior distância interna do polígono, no planejamento de rota do *Mission Planner* é permitido alterar a orientação do preenchimento para qualquer direção. Para estes testes foram mantidos a sugestão de orientação inicial do *Mission Planner*, já que sua primeira sugestão realiza a aplicação do padrão também no sentido da maior distância interna do terreno, assim como no método proposto.

Na sequência foram realizados mais 4 testes utilizando cenários reais de características semelhantes à este apresentado, sendo eles compostos por um único polígono de formato convexo.

### 5.1.2 Cenário 2

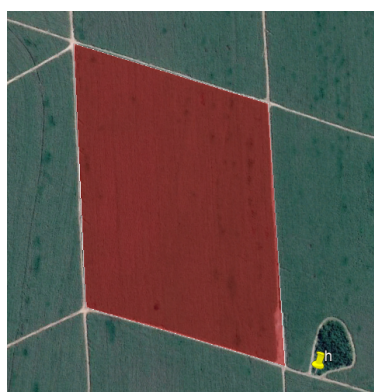
O terreno escolhido para o segundo cenário de teste pode ser observado na Figura 25(a), seguido por sua representação no algoritmo na Figura 25(b) e da rota resultante no *Mission Planner* na Figura 25(c).

Após a geração das duas rotas, utilizando a rotina de comparação desenvolvida no Matlab,

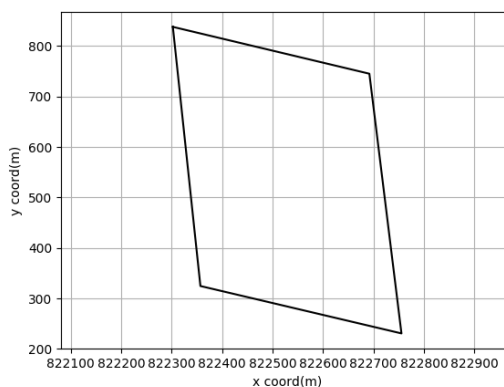
nota-se mais uma vez um grande semelhança entre as rotas propostas, como pode-se observar nas Figuras 25(d) e 25(f).

O número de curvas e distância total das rotas foram praticamente idênticos, reforçando que em casos simples como este, o método desenvolvido apresenta uma proposta de rota similar a rota gerada no *Mission Planner*, e uma convergência instantânea do algoritmo do ACO, como ilustra a Figura 25(e).

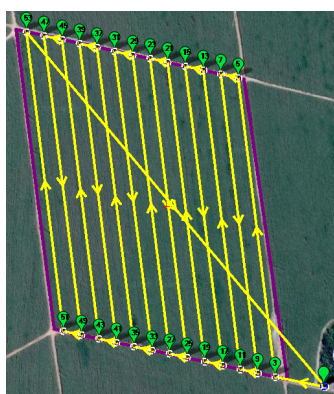
**Figura 25: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 2**



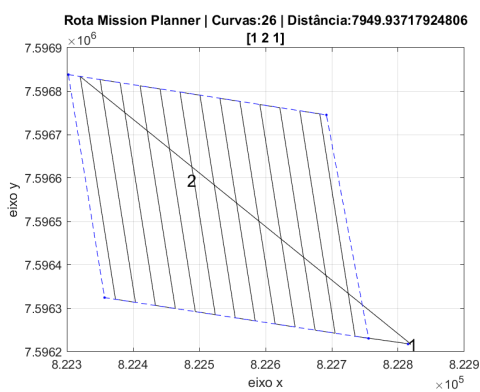
(a) Terreno selecionado no Google Earth Pro



(b) Terreno representado no algoritmo

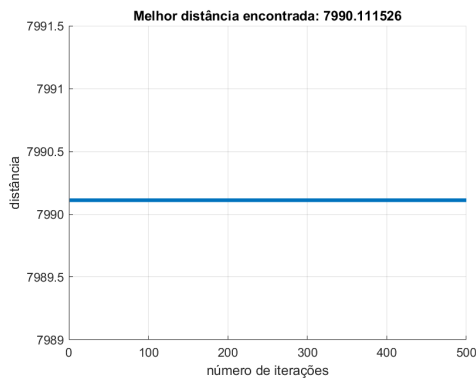


(c) Terreno preenchido com opção Survey no Mission Planner

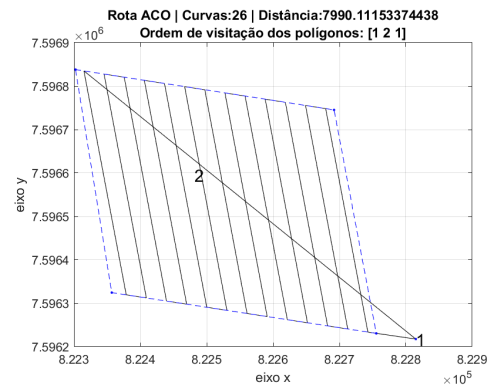


(d) Rota final do Mission Planner no Matlab





(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Na Tabela 4 observa-se através da variação percentual que a diferença da distância total da rota não chega ao valor de 1%, o que reforça a semelhança entre as duas rotas.

**Tabela 4: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 2**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	26	7949.93	2	2	1-2-1
Rota 2 - Método proposto	26	7990.11	2	2	1-2-1
Variação percentual	0	0.50			

**Fonte: Própria (2019)**

### 5.1.3 Cenário 3

No cenário 3 também foi utilizado um terreno de formato convexo, diferenciando dos terrenos dos cenários anteriores no sentido de não ser mais um polígono no formato quadrilátero, como pode ser observado na Figura 26(a) juntamente com sua representação no algoritmo retratado pela Figura 26(b) e a rota obtida no *Mission Planner*, representada na Figura 26(c).

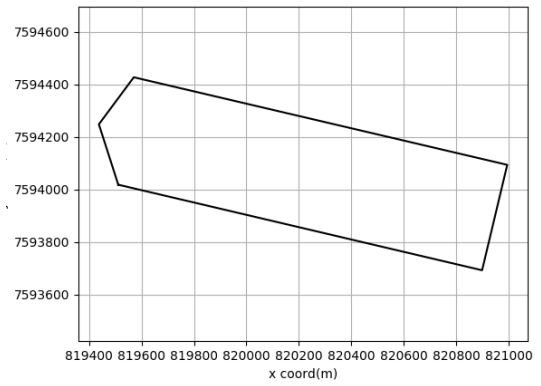
A análise gráfica das rotas representadas nas Figuras 26(d) e 26(f) indica novamente resultados similares. Sendo a única diferença visível entre as duas rotas um segmento de reta que consta na rota do *Mission Planner* ligando uma linha do percurso a outra linha mais distante, de forma para preencher pedaço de área que não seria preenchido.

O gráfico de convergência do algoritmo ACO pode ser observado na Figura 26(f).

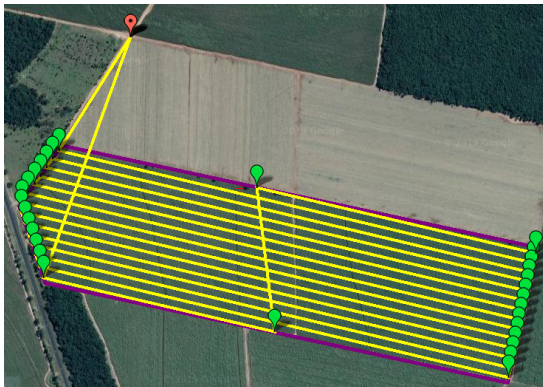
**Figura 26:** Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 3



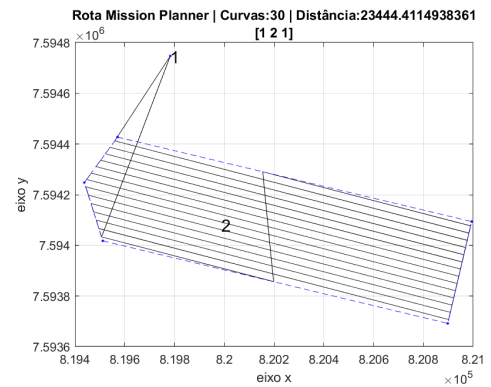
(a) Terreno selecionado no Google Earth Pro



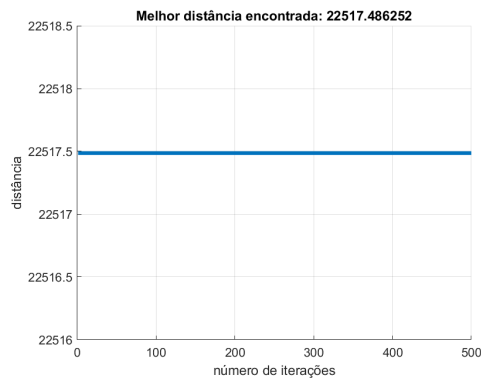
(b) Terreno representado no algoritmo



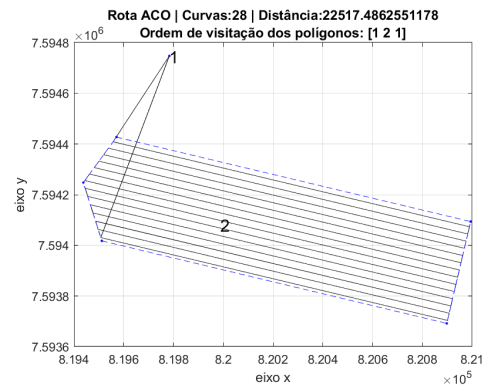
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Neste cenário, o método desenvolvido resultou em uma rota que diminui duas curvas em relação a rota proposta pelo *Mission Planner*. Na Tabela 5 é possível perceber que por conta do terreno não ter uma grande extensão de área, as duas curvas a menos na rota representou uma diminuição de 6,66% em relação a rota 1. Já quanto a distância total do trajeto, houve uma economia próxima de 4%.

**Tabela 5: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 3**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	30	23444.41	2	2	1-2-1
Rota 2 - Método proposto	28	22517.48	2	2	1-2-1
Varição percentual	-6.66	-3.95			

**Fonte: Própria (2019)**

#### 5.1.4 Cenário 4

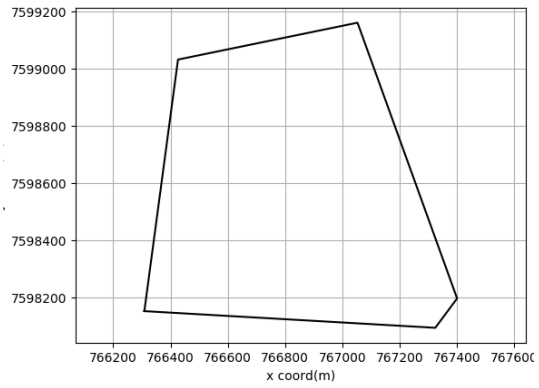
Para cenário 4 de teste foi escolhido o terreno representado nas Figuras 27(a) e 27(b). Na Figura 27(c) pode ser observado a rota resultante do planejamento de rota no *Mission Planner*.

Através das rotas resultantes observadas nas Figuras 27(d) e 27(f) nota-se que as soluções proposta foram graficamente semelhantes.

**Figura 27: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 4**



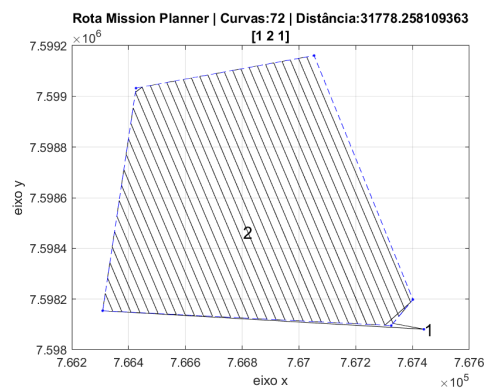
(a) Terreno selecionado no Google Earth Pro



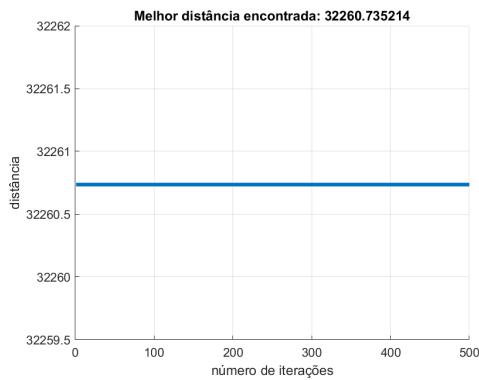
(b) Terreno representado no algoritmo



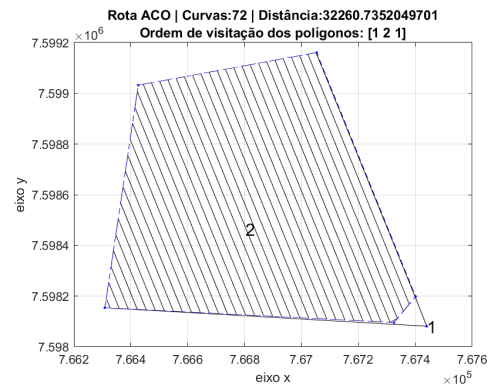
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Na Tabela 6 observa-se que apesar do número de curvas do percurso ser o mesmo nas duas rotas, o método proposto resultou em um aumento na distância total do trajeto de 1.51% em relação a rota do Mission Planner.

**Tabela 6: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 4**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	72	31778.25	2	2	1-2-1
Rota 2 - Método proposto	72	32260.73	2	2	1-2-1
Variação percentual	0	1.51			

**Fonte: Própria (2019)**

### 5.1.5 Cenário 5

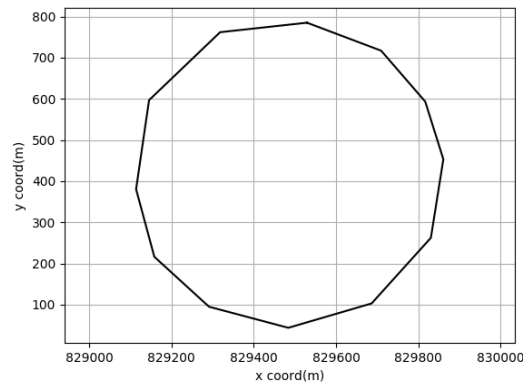
O cenário 5 tem como área de interesse um pivô de irrigação. Este terreno tem um formato circular devido a estrutura móvel utilizada para irrigação da área. O terreno foi mapeado no *Google Earth Pro* de forma a ser um polígono convexo, como ilustram as Figuras 28(a) e 28(b). Na Figura 28(c) encontra-se a rota propostas para o cenário 5 no *Mission Planner*.

Através uma análise gráfica pode ser observar nas Figuras 28(d) e 28(f), que os métodos seguiram direções diferentes de preenchimento da área, mas que apesar disto, comparando os valores da Tabela 7, o número de curvas e a distância total do percurso foram praticamente os mesmos, tendo uma diferença menor que 1% na distância total da rota.

**Figura 28:** Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 5



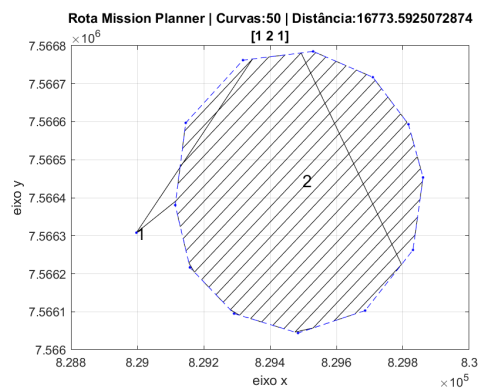
(a) Terreno selecionado no Google Earth Pro



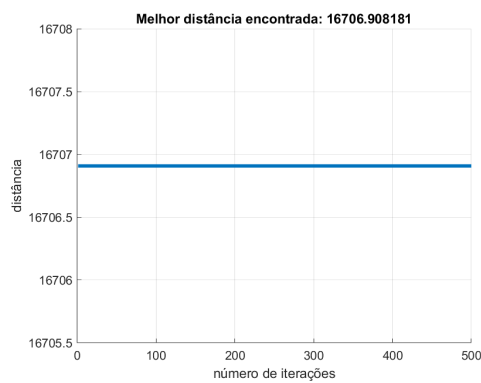
(b) Terreno representado no algoritmo



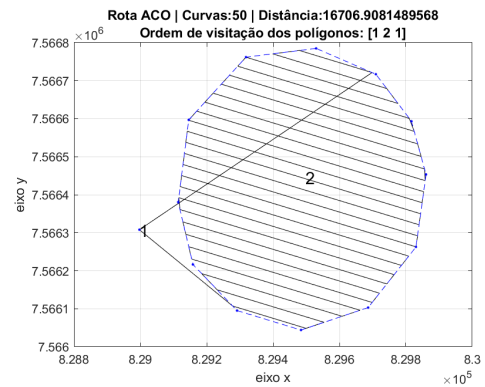
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

**Tabela 7: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 5**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	50	16773.59	2	2	1-2-1
Rota 2 - Método proposto	50	16706.90	2	2	1-2-1
Variação percentual	0	-0.39			

**Fonte: Própria (2019)**

Nestes primeiros cinco cenários de teste, que se caracterizam por conter um único terreno de formato convexo, os dois métodos apresentaram resultados muito parecidos, ou seja, os dois métodos propuseram rota similares, sem grandes diferenças entre elas. São cenários como estes os mais comuns em missões de mapeamento de área, onde o piloto desenha um grande retângulo englobando a área de interesse a ser mapeada, e com base no polígono desenhado, o programa de planejamento propõe uma rota de cobertura.

Os próximos cinco cenários testam os 2 métodos em terrenos únicos de formato côncavo.

## 5.2 Cenários de terreno único: formato côncavo

### 5.2.1 Cenário 6

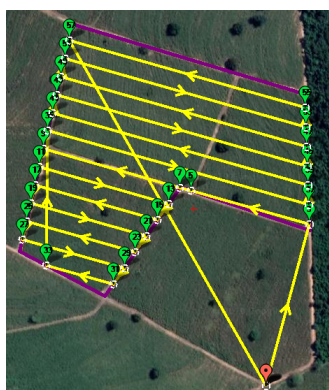
O cenário 6 é constituído por um ponto de lançamento e um terreno de formato côncavo como representado nas Figuras 29(a) e 29(b). A Figura 29(c) representa a rota resultante do planejamento no *Mission Planner*.

Por se tratar de um terreno côncavo, a primeira rotina do algoritmo proposto irá dividir o polígono côncavo em múltiplos polígonos convexos de tamanhos menores. Em uma análise gráfica, observa-se na Figura 29(f) que o algoritmo dividiu a área de entrada em duas subáreas, sendo assim, apesar de originalmente ser uma única área de interesse, o algoritmo apresenta um cenário final constituído por: um ponto de lançamento e duas áreas de interesse.

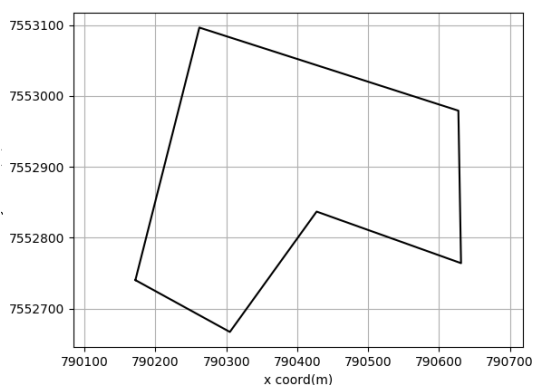
Visualmente nota-se que cada um dos métodos aplicou o padrão *boustrophedon* em sentidos diferentes. Na rota do *Mission Planner* o padrão foi aplicado em um mesmo sentido sobre a área inteira, como mostra a Figura 29(d). Já no método proposto, cada subárea resultante da decomposição teve um sentido de aplicação próprio, de forma a minimizar o número de curvas em cada uma.

Neste caso por existir duas subáreas para ser tratadas, o algoritmo do ACO precisou avaliar as diferentes combinações existentes resultantes das quatro possibilidades de cobertura de cada terreno. Apesar disso, por ser um número baixo de combinações o resultado foi instantâneo, como ilustra o gráfico de convergência na Figura 29(e).

**Figura 29:** Fases do processo de comparação de rota entre *Mission Planner* e método proposto para o cenário 6



(a) Terreno seleccionado no Google Earth Pro

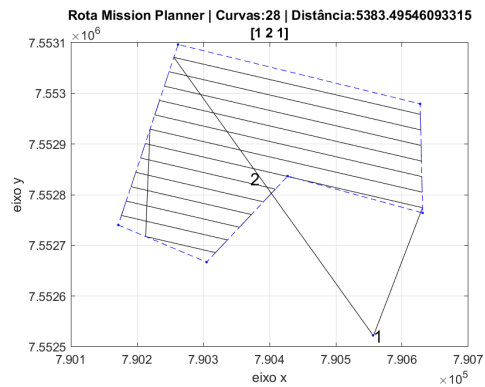


(b) Terreno representado no algoritmo

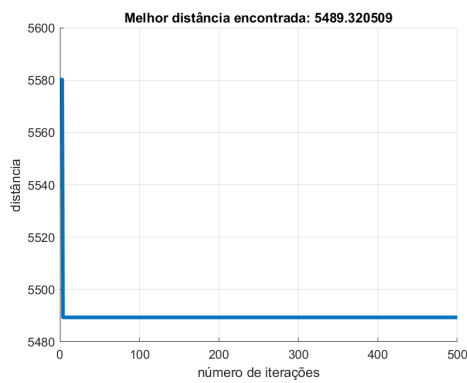




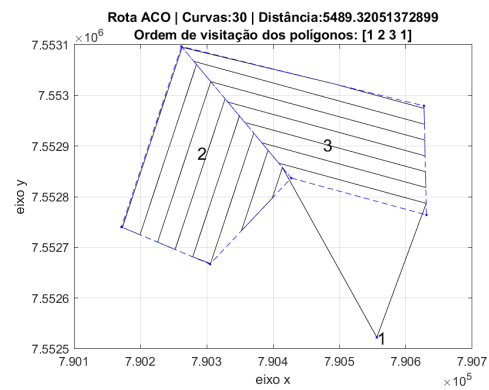
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Através de uma análise mais detalhada utilizando a Tabela 8, observa-se que o fato de minimizar o número de curvas em cada subárea separadamente, na solução como um todo, o método proposto acabou por aumentar um segmento de reta no trajeto total, o que em porcentagem, por conta da área ser pequena, significou um aumento de 7.14% no número de curvas, e um aumento de 1.96% na distância total do trajeto. Por conta da geração de subáreas, a ordem de visitação dos terrenos do método proposto ficou sendo 1-2-3-1.

**Tabela 8: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 6**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	28	5383.49	2	2	1-2-1
Rota 2 - Método proposto	30	5489.32	2	3	1-2-3-1
Varição percentual	7.14	1.96			

**Fonte: Própria (2019)**

### 5.2.2 Cenário 7

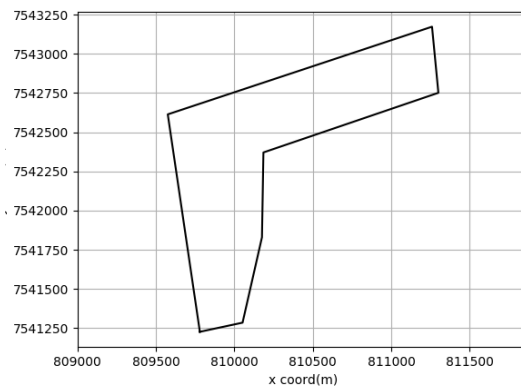
No cenário 7 procurou-se uma área de proporções maiores mas mantendo a similaridade de características com o cenário anterior. O cenário escolhido está representado nas Figuras 30(a) e 30(b), enquanto na Figura 30(c) se encontra o a rota proposta pelo *Mission Planner*.

Pela análise dos gráficos das Figuras 30(d) e 30(f) observa-se que visualmente os resultados foram similares aos obtidos no teste do cenário anterior.

**Figura 30: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 7**



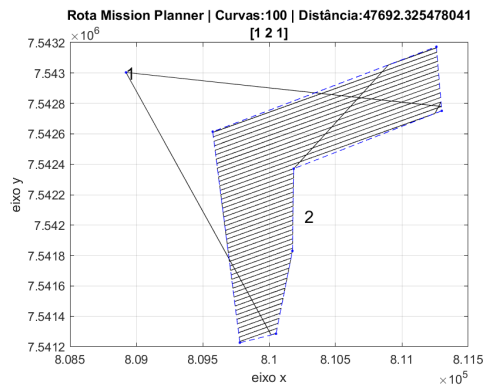
(a) Terreno selecionado no Google Earth Pro



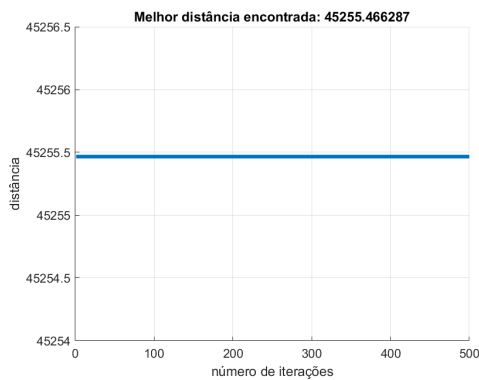
(b) Terreno representado no algoritmo



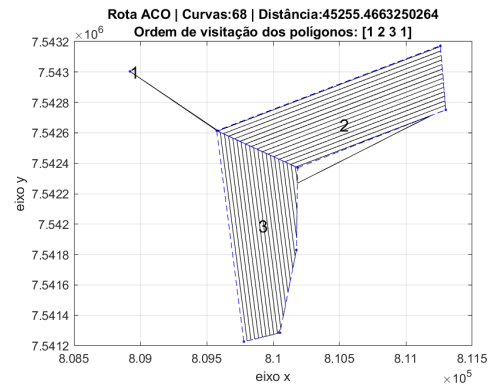
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Observando a Tabela 9 nota-se que o método proposto conseguiu diminuir tanto o número de curvas do percurso quanto a distância total, em 32% e 5.10% respectivamente. Apesar de graficamente as rotas propostas serem muito similares as do cenário 6, neste caso, a área do terreno possui uma maior dimensão. Para este caso, o fato de decompor o terreno e aplicar o padrão *boustrophedon* em orientações diferentes para cada subárea teve maior relevância, evitando a realização de diversas curvas desnecessária comparado a solução do *Mission Planner*.

**Tabela 9: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 7**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	100	47692.32	2	2	1-2-1
Rota 2 - Método proposto	68	45255.46	2	3	1-2-3-1
Varição percentual	-32	-5.10			

**Fonte: Própria (2019)**

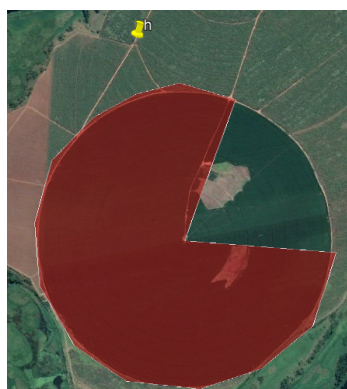
### 5.2.3 Cenário 8

Para o cenário 8 foi utilizado como área de interesse um pivô de irrigação. Diferente do pivô apresentado no cenário 5, neste caso, somente parte dele é de interesse em ser mapeado, podendo ser representado como um polígono côncavo como representado nas Figuras 31(a) e 31(b).

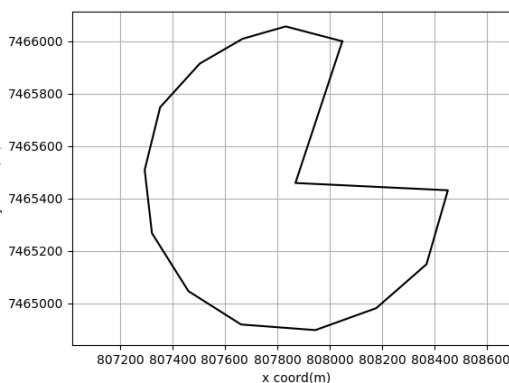
Pelo fato de se tratar de um terreno côncavo, a rotina de decomposição do método proposto atuou sobre o terreno para decompô-lo em 2 subáreas de formatos convexos.

Visualmente, observando as Figuras 31(d) e 31(f), não é perceptível qual método se saiu melhor neste cenário, sendo necessário uma análise mais detalhada.

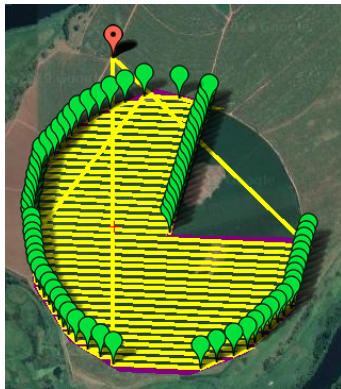
**Figura 31: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 8**



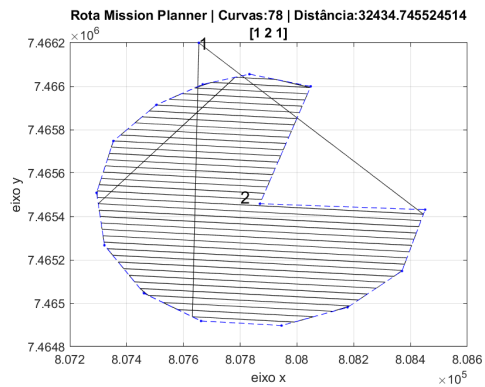
(a) Terreno selecionado no Google Earth Pro



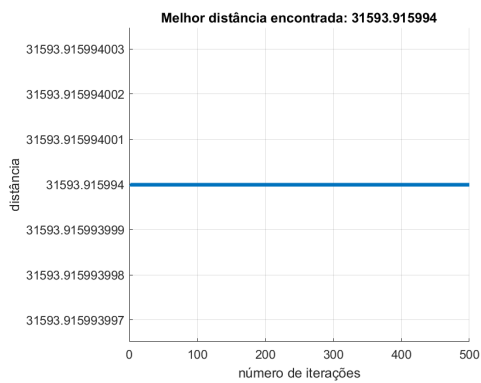
(b) Terreno representado no algoritmo



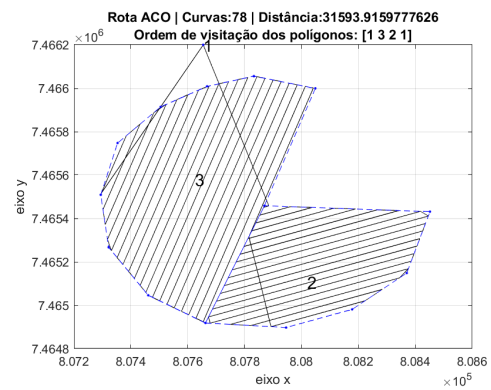
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Na Tabela 10 observa-se que o número de curvas das duas rotas foram os mesmos, e que a rota do método proposto resultou em uma menor distância total de percurso, devido ao fato de existir um segmento de linha a mais na rota do *Mission Planner*, com a função de ligar uma linha de voo com outra mais distante. Com exceção desta diferença, as duas rotas tiveram resultados semelhantes.

**Tabela 10: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 8**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	78	32434.74	2	2	1-2-1
Rota 2 - Método proposto	78	31593.91	2	3	1-3-2-1
Varição percentual	0	-2.59			

Fonte: Própria (2019)

### 5.2.4 Cenário 9

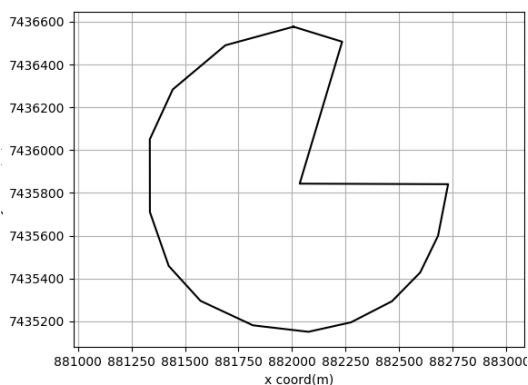
No cenário 9 foi escolhido um terreno de maior dimensão, mas com as mesmas características do cenário 8, como demonstra as Figuras 32(a) e 32(b).

Porém, mesmo contando com uma área ligeiramente maior, os 2 métodos apresentaram resultados praticamente idênticos quanto ao número de curvas e distância total do percurso, como pode ser observado tanto nas Figuras 32(d) e 32(f) quanto na Tabela 11.

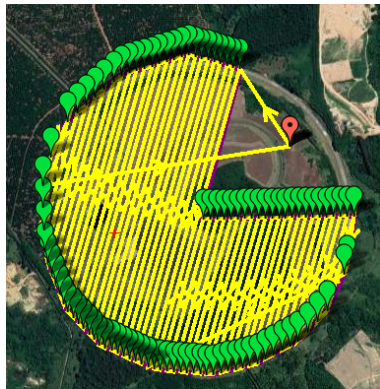
**Figura 32: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 9**



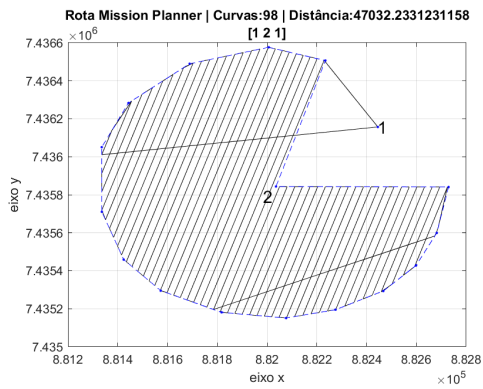
(a) Terreno selecionado no Google Earth Pro



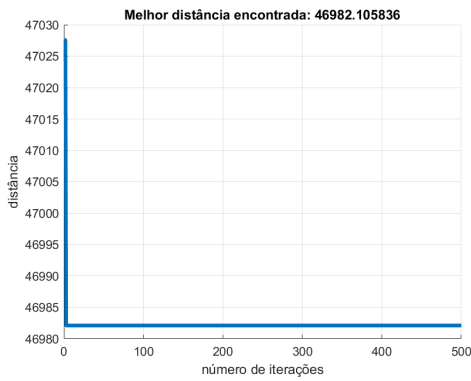
(b) Terreno representado no algoritmo



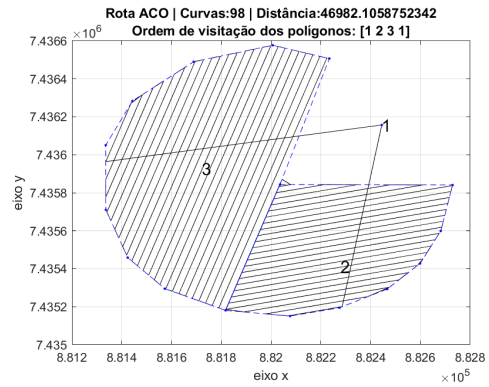
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Tabela 11: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 9

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	98	47032.23	2	2	1-2-1
Rota 2 - Método proposto	98	46982.10	2	3	1-2-3-1
Varição percentual	0	-0.10			

Fonte: Própria (2019)

### 5.2.5 Cenário 10

O cenário 10 é composto pelo ponto de lançamento do VANT e pelo o terreno de formato concavo apresentado na Figura 33(a). A representação computacional do terreno pode ser ob-

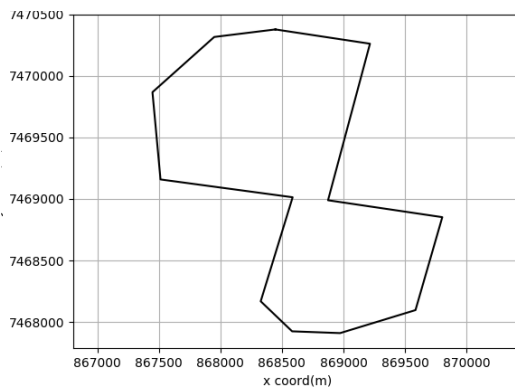
servada na Figura 33(b), enquanto a a solução proposta pelo *Mission Planner* ficou registrada na Figura 33(c).

Neste cenário as rotas resultantes apresentaram diferenças entre si, como pode ser observado nas Figuras 33(d) e 33(f). Cada um dos métodos aplicou o padrão *boustrophedon* em diferentes orientações, além disso no método proposto por se tratar de um polígono de formato côncavo, houve decomposição de terreno em 2 subáreas de formato convexo.

**Figura 33: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 10**



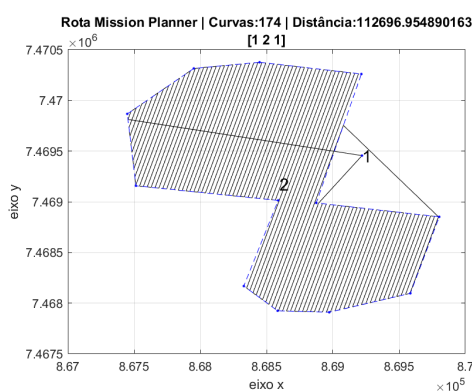
(a) Terreno selecionado no Google Earth Pro



(b) Terreno representado no algoritmo

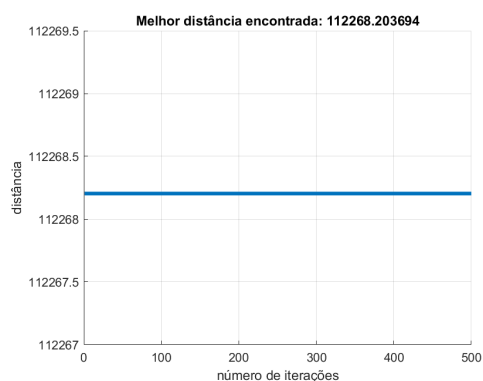


(c) Terreno preenchido com opção Survey no Mission Planner

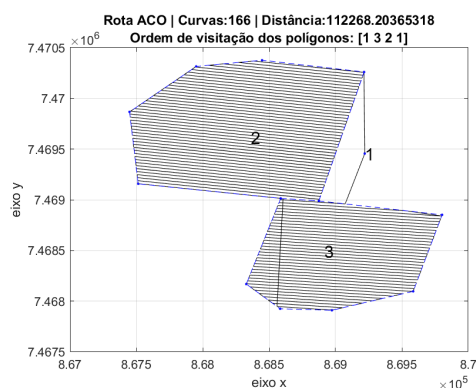


(d) Rota final do Mission Planner no Matlab





(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Através da Tabela 12 percebe-se que o método proposto diminuiu em 4.59% o número de curvas do percurso, apesar da distancia total do percurso ser praticamente a mesma nas duas rotas.

**Tabela 12: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 10**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	174	112696.95	2	2	1-2-1
Rota 2 - Método proposto	166	112268.20	2	3	1-3-2-1
Variação percentual	-4.59	-0.38			

**Fonte: Própria (2019)**

Observando os terrenos côncavos apresentados nos cenário de 5 a 10, percebe-se que o fato de decompor o terreno em subáreas pode ajudar ou atrapalhar dependendo do caso. Deste modo decompor terreno é uma técnica que não pode ser tomado como regra para todos os casos, seu uso deve ser avaliado.

## 5.3 Cenários com cinco terrenos

### 5.3.1 Cenário 11

À partir do cenário 11 os casos de testes passam a ter mais de um terrenos que devem ser considerados na proposta de rota. Este cenário 11 conta com cinco terreno iniciais como ilustra as Figuras 34(a) e 34(b). Dentre os cinco, três são côncavos, desta forma, ao serem

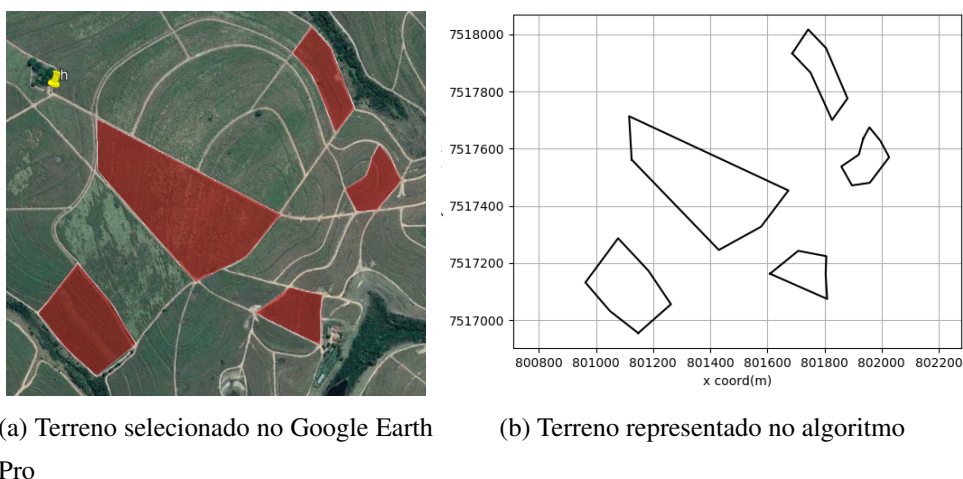
processados pelo método proposto, após a fase de decomposição, o total de cinco terrenos iniciais se torna oito. Para cada área processada no método deste trabalho, existem quatro alternativas de cobertura diferentes, sendo assim, para este caso com oito áreas, sendo assim totaliza-se 65536 combinações diferentes que o algoritmo do ACO deve tratar.

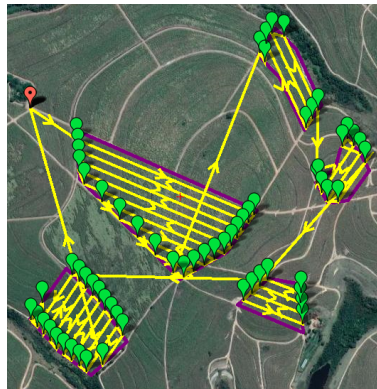
Com base neste fato, nota-se a importância da utilização da meta-heurística do ACO no cenários com um número maior terrenos, já que considerar todas as possíveis combinações através de algoritmos de força bruta passar a ficar computacionalmente custoso.

Na Figura 34(e) pode-se observar o gráfico de convergência do algoritmo de ACO. Neste caso, apesar de não ter sido uma solução instantânea devido ao número de combinações para se tratar, devido a parametrização do algoritmo, em menos de 100 iterações foi possível apresentar uma proposta de rota com uma distância total menor que a rota do *Mission Planner*

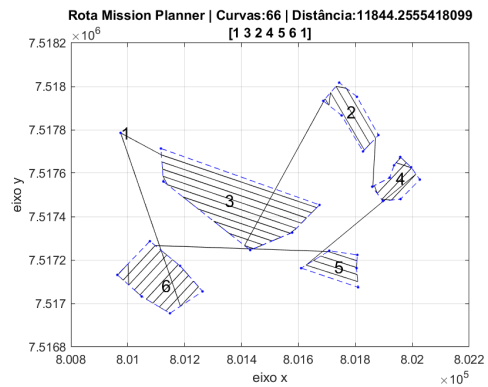
Outro fato que pode ser percebido é que conforme o cenário cresce e o número de terrenos aumenta, o algoritmo do *Mission Planner* e o desenvolvido neste trabalho passam a apresentar propostas de rotas cada vez mais diferentes entre si, como retratado nas Figuras 34(d) e 34(f).

**Figura 34: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 11**

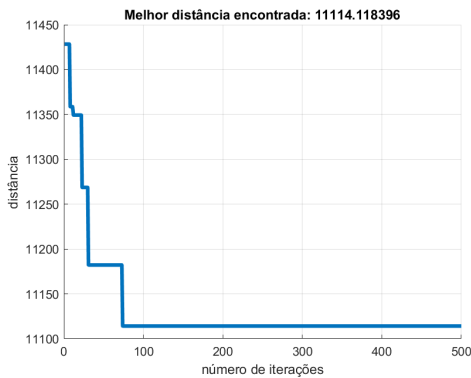




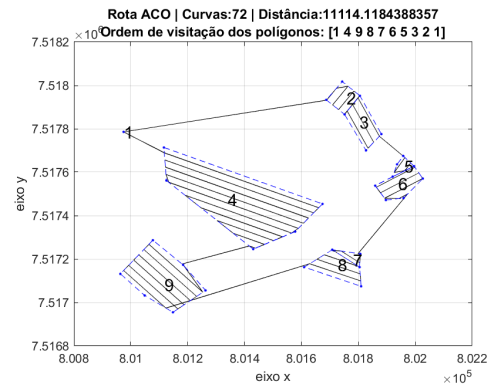
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

**Fonte: Própria (2019)**

Através da tabela 13 observa-se que o método do trabalho gerou uma rota com maior número de curvas que o do *Mission Planner*, cerca de 9% a mais. Apesar disso o método proposto apresentou uma diminuição de -6.16% na distância total em comparação com o *Mission Planner*.

Como já citado anteriormente, o fato de sempre decompor terrenos côncavos em côncavos, não é uma estratégia que sempre diminui o número de curvas no percurso, e neste caso houve um aumento na quantidade de curvas do percurso. Apesar disto, ainda foi possível diminuir a distância total do percurso, devido ao motivo de que o algoritmo de planejamento de rota do *Mission Planner* não considerou quais seriam os melhores pontos de saída e de entrada de cada terreno, nem tanto pouco considerou qual seria a melhor ordem de visitação. Percebeu-se que a rota proposta por ele seguiu a ordem de inserção dos polígonos para serem mapeados.

**Tabela 13: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 11**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	66	11844.25	6	6	1-3-2-4-5-6-1
Rota 2 - Método proposto	72	11114.11	6	9	1-4-9-8-7-6-5-3-2-1
Varição percentual	9.09	-6.16			

**Fonte: Própria (2019)**

### 5.3.2 Cenário 12

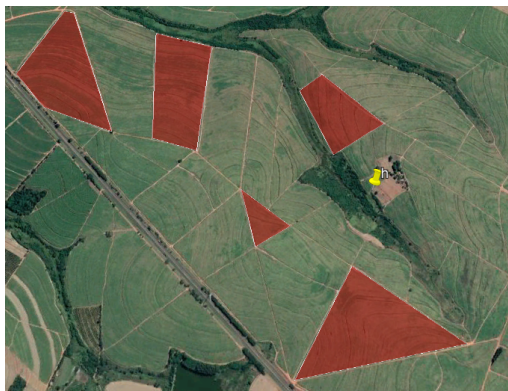
No cenário 12 foram escolhidos somente terrenos de formato côncavos para uma comparação visando avaliar o impacto isolado da otimização de ordem de visitação. Os terrenos deste cenário de teste podem ser visualizados nas Figuras 35(a) e 35(b).

Para planejar a rota deste cenário no *Mission Planner*, como representado na Figura 35(c), tomou-se o devido cuidado de inserir os terreno para serem processados em uma ordem visualmente otimizada, se baseando na fato de que por conta do programa não tratar a ordem de visitação esta preocupação ficaria a cargo do piloto.

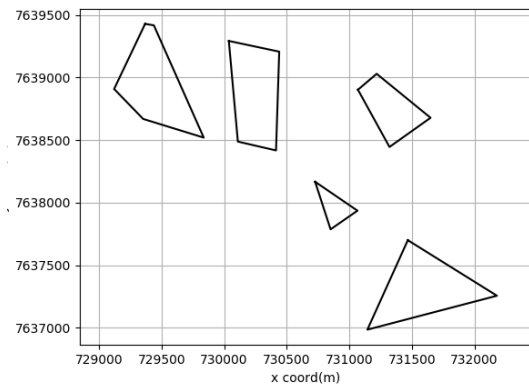
Quanto a solução dada pelo método proposto, a convergência do algoritmo de ACO mais uma vez não precisou que mais de 100 ciclos para encontrar uma rota de menor distância que a do *Mission Planner*.

Após a execução da rotina de análise no Matlab, percebe-se que as duas rotas consideraram os mesmo vizinhos para cada terreno, como pode-se observar nas Figuras 35(d) e 35(f), trocando somente o sentido da visitação, enquanto o *Mission Planner* parte do ponto 1 em direção ao terreno 6 e ao final retorna pelo 3, o método proposto parte do ponto 1 em direção ao terreno 3 e ao final retorna pelo terreno 6.

**Figura 35: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 12**



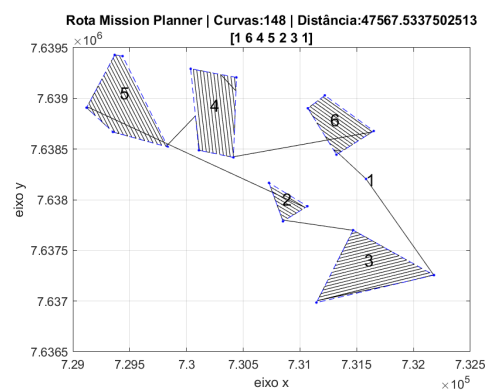
(a) Terreno selecionado no Google Earth Pro



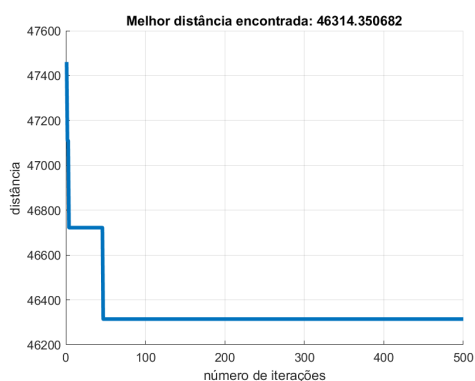
(b) Terreno representado no algoritmo



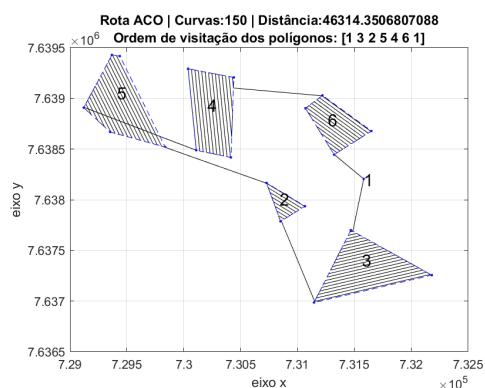
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Através da Tabela 14 observa-se que o método proposto teve um segmento de reta a mais comparado a rota do *Mission Planner*, o que gerou 2 curvas a mais no percurso. Em contrapartida, o método proposto economizou -2.63% na distância total do trajeto.

**Tabela 14: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 12**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	148	47567.53	6	6	1-6-4-5-2-3-1
Rota 2 - Método proposto	150	46314.35	6	6	1-3-2-5-4-6-1
Variação percentual	1.35	-2.63			

Fonte: Própria (2019)

### 5.3.3 Cenário 13

O cenário de teste número 13, segue as características do cenário anterior, sendo composto por o ponto de lançamento do VANT, mais 5 terreno de formato convexo, a diferença é que a dimensão deste cenário é maior comparado ao anterior. A distribuição dos terrenos podem ser observados nas Figuras 36(a) e 36(b). Foi tomado o devido cuidado de não inserir os terrenos para processar em uma ordem aleatória no *Mission Planner*. O resultado do processamento no *Mission Planner* pode ser observado na Figura 36(c).

Através das Figuras 36(d) e 36(f) observa-se que as rotas propostas no dois método são graficamente similares.

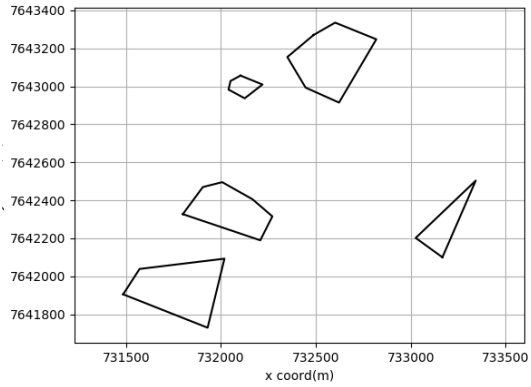
Chama-se atenção para o gráfico de convergência do algoritmo de ACO, que encontrou uma

distância de rota pequena logo nas primeiras iterações, como pode-se observar na Figura 36(e).

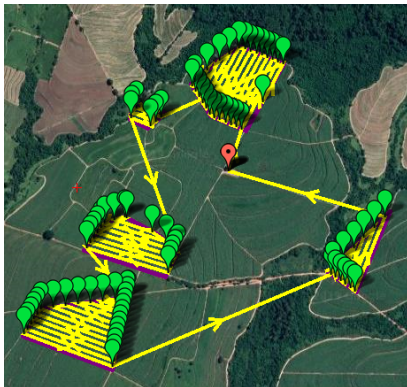
**Figura 36:** Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 13



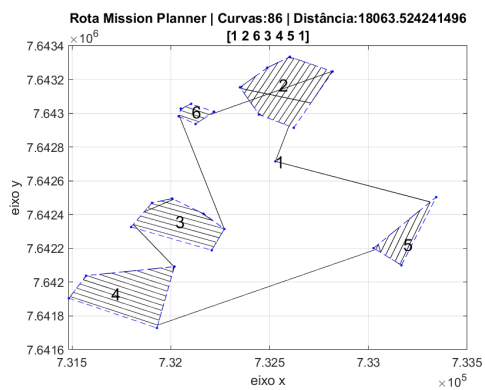
(a) Terreno selecionado no Google Earth Pro



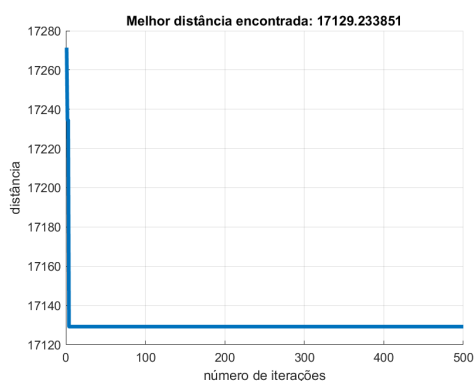
(b) Terreno representado no algoritmo



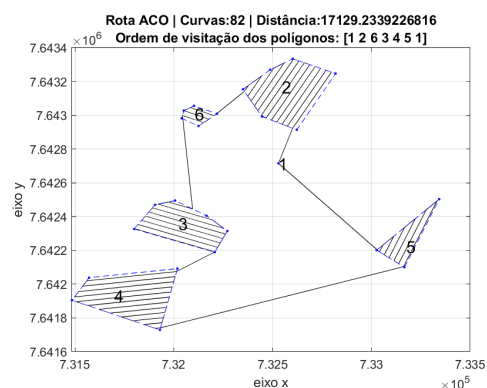
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Em uma análise mais detalhada através da Tabela 15, percebe-se que no método proposto, diminuindo dois seguimentos de retas, eliminou 4 curvas no trajeto, o que representou uma diminuição de -4.65 no número de curvas, enquanto em relação a distância total do percurso a economia foi de -5.17%. Mesmo a ordem de visitação tenha sido a mesma nos métodos para este cenário, o método proposto analisa 4 alternativas de cobertura diferentes em cada terreno, onde cada uma possui seu ponto de entrada e saída do terreno, e dentre as 4 alternativas o algoritmo escolhe a que resulta na menor distância no conjunto total de combinações. Já no método do *Mission Planner* a rota interna do próximo terreno parte do ponto de saída do terreno atual, sem qualquer análise adicional.

**Tabela 15: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 13**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	86	18063.52	6	6	1-2-6-3-4-5-1
Rota 2 - Método proposto	82	17129.23	6	6	1-2-6-3-4-5-1
Variação percentual	-4.65	-5.17			

Fonte: Própria (2019)

### 5.3.4 Cenário 14

O cenário 14 é composto por 5 áreas convexas como mostrado nas Figuras 37(a) e 37(b).

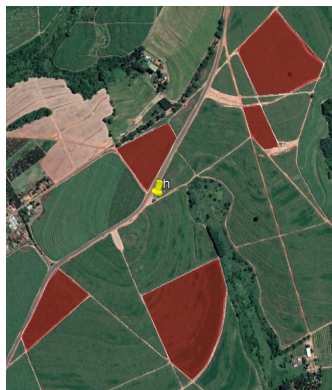
Para este cenário a gráfico de convergência, representando na Figura 37(e), teve sua última atualização de melhor rota por volta da iteração de número 200, porém assim como nos casos



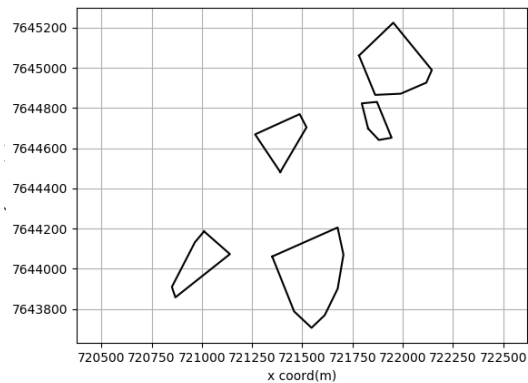
anteriores apresenta seus maiores ganhos dentro das 100 primeiras iterações, remetendo à uma parametrização do algoritmo com uma característica muito forte de exploração.

Pela análise gráfica das Figuras 37(d) e 37(f) percebe-se as duas rotas propostas possuem os a mesma ordem de visitação, porém nota-se que os pontos de entrada e saída dos terrenos são diferentes.

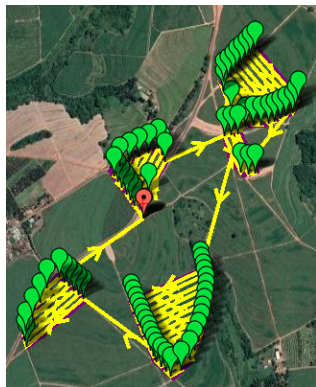
**Figura 37: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 13**



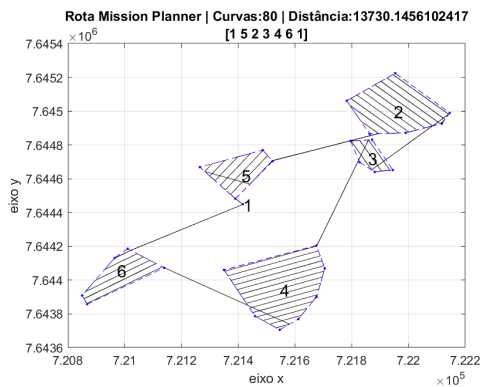
(a) Terreno selecionado no Google Earth Pro



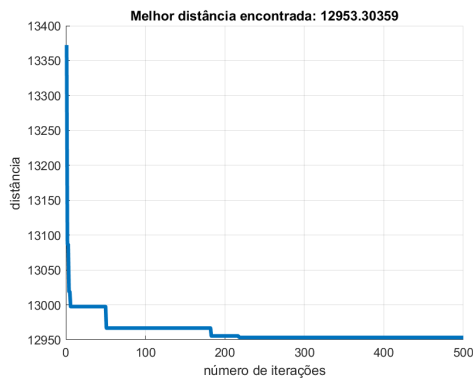
(b) Terreno representado no algoritmo



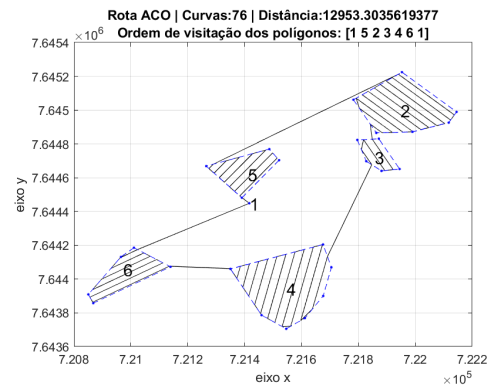
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Utilizando a Tabela 16 é possível verificar que o método proposto conseguiu uma diminuição de 5% no número de curvas do percurso, e 5.65% na distância total.

Pensando em uma situação real, o valor economizado não representa muito perto do total da rota, pois estaria poupando energia de 4 curvas e mais 800 metros em linha reta, porém considerando que o modelo do VANT é um multirrotor, e que tem uma autonomia bem limitada perto dos modelos de asa fixa, o valor economizado representa uma quantidade significativa de área a mais a ser mapeada.

**Tabela 16: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 14**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	80	13730.14	6	6	1-5-2-3-4-6-1
Rota 2 - Método proposto	76	12953.30	6	6	1-5-2-3-4-6-1
Varição percentual	-5	-5.65			

Fonte: Própria (2019)

### 5.3.5 Cenário 15

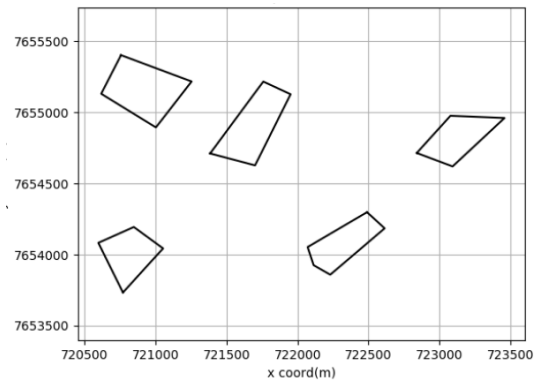
O cenário de número 15 é composto pelos terrenos apresentados nas Figuras 38(a) e 38(b), sendo todos polígonos de formato convexo, assim como os 3 últimos cenários apresentados. Mais um vez os polígonos foram inseridos para processamento no *Mission Planner* em uma ordem otimizada visualmente, para que desta forma seja feita uma comparação equilibrada, de forma que diminuindo as diferenças de comportamento nos pontos já mapeados, possam ficar

evidente outros detalhes que estavam encobertos.

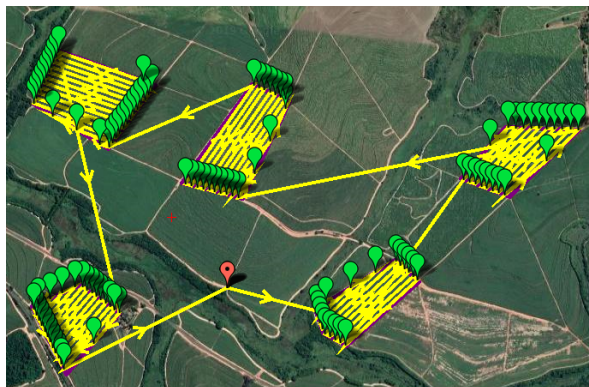
**Figura 38:** Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 14



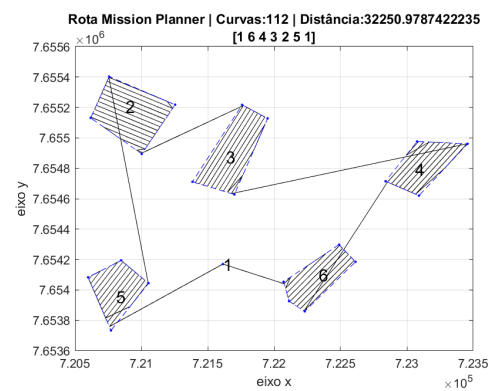
(a) Terreno selecionado no Google Earth Pro



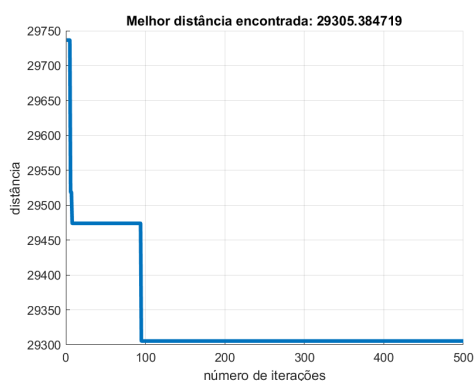
(b) Terreno representado no algoritmo



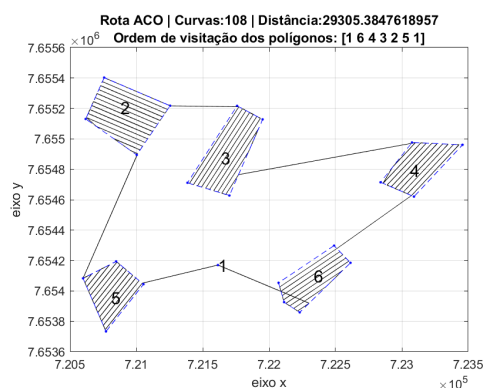
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Na Tabela 17 percebe-se que o ganho do método proposto sobre o método do *Mission Planner* foi de 4 curvas, representando -3.57%, e de cerca de 9% na distância total.

**Tabela 17: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 15**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	112	32250.97	6	6	1-6-4-3-2-5-1
Rota 2 - Método proposto	108	29305.38	6	6	1-6-4-3-2-5-1
Variação percentual	-3.57	-9.13			

Fonte: Própria (2019)

Nestes cenário de 11 a 15 foi priorizado cenários somente com áreas convexas, para uma análise focada no ganho proporcionado pelo algoritmo de ACO, já que nos teste de cenário de 5 a 10 observou-se que a decomposição de terreno nem sempre traz ganhos.

Nota-se também que o algoritmo de ACO não tem nenhuma influência sobre o número de curvas da rota, sendo esta variável dependente do resultado da decomposição do terreno e da orientação de preenchimento utilizando o padrão *boustrophedon*.

Como ponto positivo para o método proposto, pode-se observar o fato de que mesmo colocando os dois métodos para visitarem os terrenos em um mesma sequência, o método proposto considera 4 possíveis alternativas de rotas internas em cada terreno, com seus respectivos pontos de entrada e saída, enquanto o método do *Mission Planner* simplesmente interliga a saída de um terreno à entrada do próximo terreno.

Outro ponto observado foi que a configuração dos parâmetros do algoritmo de ACO possi-

bilitaram um convergência rápida nos cenários testados, raramente sendo necessário mais que 150 iterações para apresentar uma solução aceitável.

Cada um dos cenário de 16 à 20 serão constituídos pelo ponto de lançamento do VANT mais 10 terrenos diferentes. Considerando que o método proposto considera 4 alternativas de rota para cada terreno, no total são 1048576 combinações diferentes de rotas que o algoritmo ACO deve tratar para um cenário deste nível.

## 5.4 Cenários com dez terrenos

### 5.4.1 Cenário 16

No cenário 16, representado pelas Figuras 39(a) e 39(b), considerando a distribuição dos 10 terrenos e do ponto de lançamento do VANT, identificar visualmente uma ordem de visita otimizada já não é uma tarefa instantânea, pois este cenário possui uma complexidade maior comparado aos cenários anteriores com menor número de terrenos.

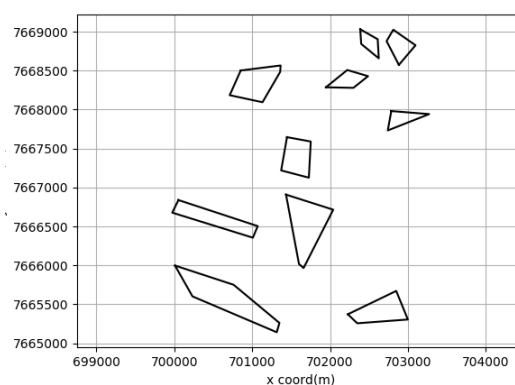
Em cenário como estes, onde o piloto não tem certeza de qual ordem de visita seria a mais otimizada, é que o algoritmo proposto pode trazer maiores benefícios.

Observa-se que mesmo neste cenário de 1048576 combinações diferentes de rotas, o algoritmo de ACO não precisou mais de 200 iterações para apresentar uma solução que melhora a rota proposta pelo *Mission Planner*, tendo seus maiores ganhos exploratórios nas 100 primeiras iterações, como pode-se observar na Figura 39(e).

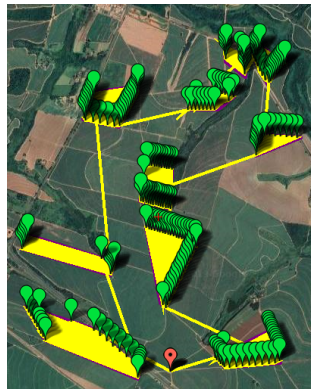
**Figura 39:** Fases do processo de comparação de rota entre *Mission Planner* e método proposto para o cenário 15



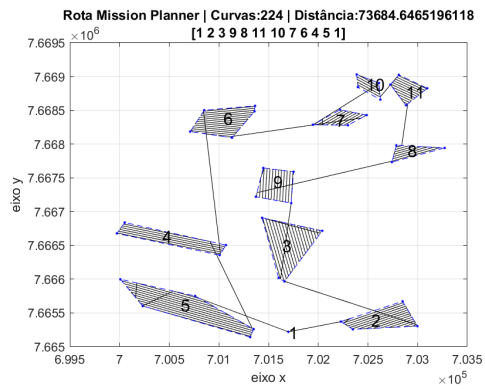
(a) Terreno seleccionado no Google Earth Pro



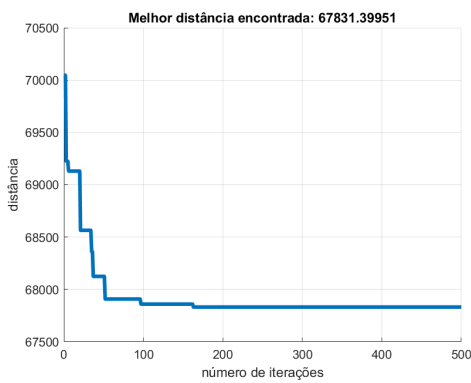
(b) Terreno representado no algoritmo



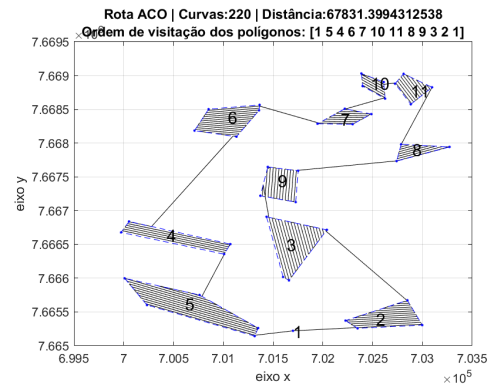
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Se tratando de um cenário somente com terreno convexos, observa-se na Tabela 18 que o número de curvas das rotas são similares entre os 2 métodos. Por outro lado, o método proposto economiza 7.94% de distância total percorrida.

**Tabela 18: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 16**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	224	73684.64	11	11	1-2-3-9-8-11-10-7-6-4-5-1
Rota 2 - Método proposto	220	67831.39	11	11	1-5-4-6-7-10-11-8-9-3-2-1
Varição percentual	-1.78	-7.94			

Fonte: Própria (2019)

### 5.4.2 Cenário 17

O cenário 17 segue as mesmas características do cenário 16, sendo composto por 10 terrenos de formato convexo como retratado nas Figuras 40(a) e 40(b).

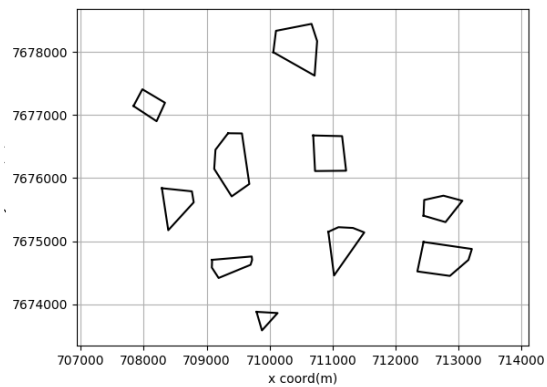
Pela análise gráfica realizada com base nas Figuras 40(d) e 40(f) percebe-se que as rotas são similares visualmente, com exceção dos terrenos vizinhos do ponto de lançamento do VANT, ponto 1, e dos pontos de entrada e saída de cada terreno.

Para este cenário o gráfico de convergência apresentou um comportamento diferente dos anteriores, apesar de nas 100 primeiras iterações já ter atingido uma proposta de rota boa que otimiza a distância em relação ao *Mission Planner*, como pode ser observado na Figura 40(e), apresentou novamente outra melhora de resultado por volta da iteração 500.

**Figura 40:** Fases do processo de comparação de rota entre *Mission Planner* e método proposto para o cenário 16



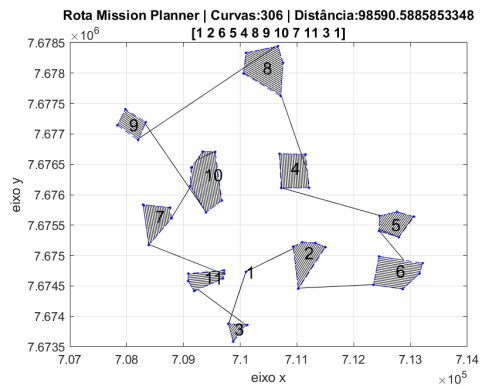
(a) Terreno selecionado no Google Earth Pro



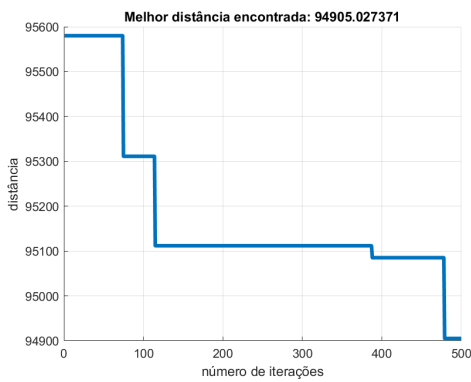
(b) Terreno representado no algoritmo



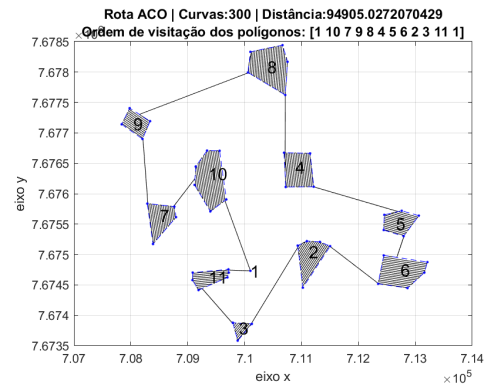
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Através da tabela 19 nota-se que os resultados não foram muito diferentes do cenário anterior, o método proposto diminuiu em 1.96% o número de curvas e 3.73% a distância total do percurso.

**Tabela 19: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 17**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	306	98590.58	11	11	1-2-6-5-4-8-9-10-7-11-3-1
Rota 2 - Método proposto	300	94905.02	11	11	1-10-7-9-8-4-5-6-2-3-11-1
Varição percentual	-1.96	-3.73			

Fonte: Própria (2019)

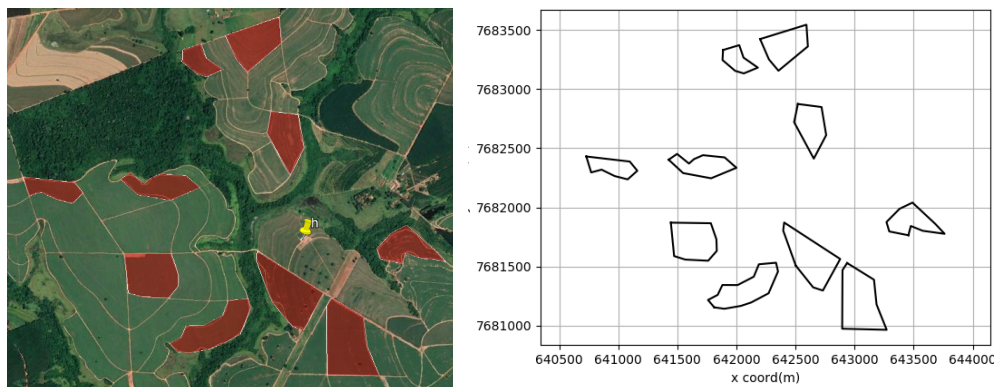


### 5.4.3 Cenário 18

No cenário 18 foi retomado o uso de terrenos côncavos para efeito de comparação de resultados com os cenários anteriores. Os terrenos escolhidos podem ser observados nas Figuras 41(a) e 41(b).

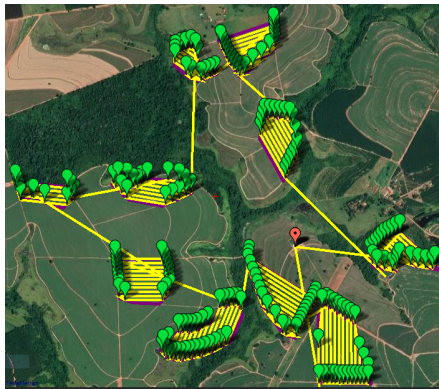
Em cenários com áreas côncavas percebe visualmente uma diferença, onde o método proposto resulta em uma rota com um número maior de terrenos que de início, como pode ser observado comparando as Figuras 41(d) e 41(f). Neste caso, o método proposto iniciou o planejamento de rota com 11 terrenos e terminou com 19. Isto é resultante do processo de decomposição de áreas côncavas em áreas convexas menores. Assim como já notado anteriormente, a decomposição de terreno pode funcionar positivamente em alguns casos, mas prejudicar em outros, como foi este caso em que o número de curvas do trajeto aumentou.

**Figura 41: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 18**

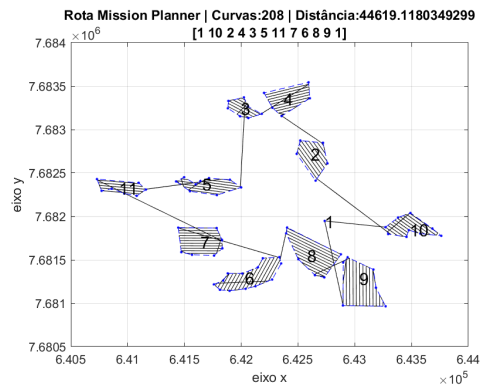


(a) Terreno selecionado no Google Earth Pro

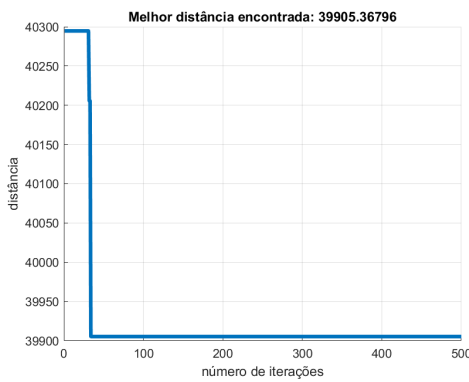
(b) Terreno representado no algoritmo



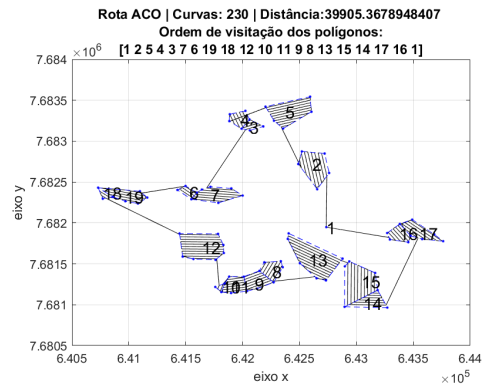
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

Apesar do número de curvas maior, o método proposto foi capaz de diminuir em 10.56% a distância total da rota, como observado na Tabela 20

**Tabela 20: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 18**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	208	44619.11	11	11	1-10-2-4-3-5-11-7-6-8-9-1
Rota 2 - Método proposto	230	39905.36	11	19	1-2-5-4-3-7-6-19-18-12-10-11-9-8-13-15-14-17-16-1
Variação percentual	10.57	-10.56			

Fonte: Própria (2019)

### 5.4.4 Cenário 19

O cenário de teste 19 é composto por terrenos côncavos e convexos assim como o cenário 18. Os terrenos escolhidos para o cenário 19 podem ser observados nas Figuras 42(a) e 42(b).

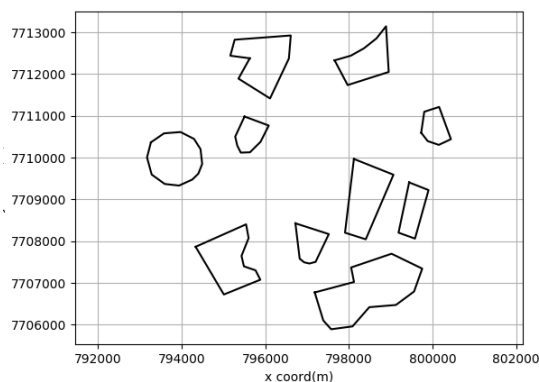
Através das Figuras 42(d) e 42(f), já na análise visual, nota-se diferenças entre as rotas dos 2 métodos, mas somente através da Tabela 21 que é possível observar que mais uma vez o método proposto aumentou o número de curvas do percurso e diminuiu a distância total.

Neste cenário o algoritmo ACO convergiu próximo a iteração 300, como pode ser observado na Figura 42(e).

**Figura 42: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 19**



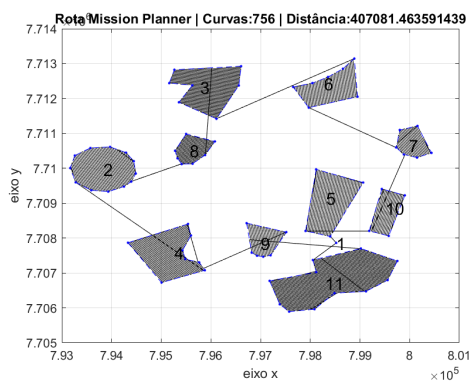
(a) Terreno selecionado no Google Earth Pro



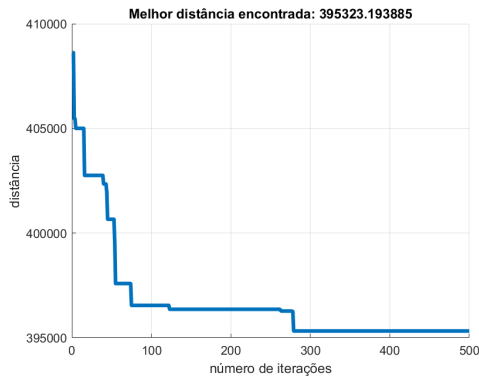
(b) Terreno representado no algoritmo



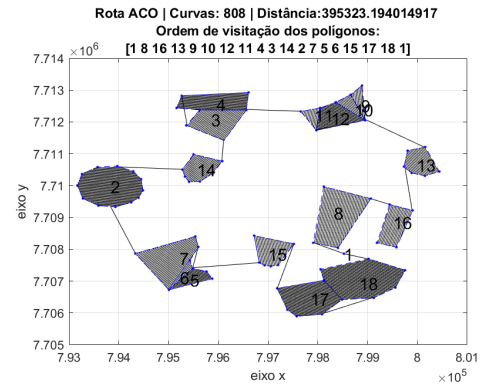
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

**Tabela 21: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 19**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	756	407081.46	11	11	1-5-10-7-6-3-8-2-4-9-11-1
Rota 2 - Método proposto	808	395323.19	11	18	1-8-16-13-9-10-12-11-4-3-14-2-7-5-6-15-17-18-1
Variação percentual	6.87	-2.88			

Fonte: Própria (2019)

### 5.4.5 Cenário 20

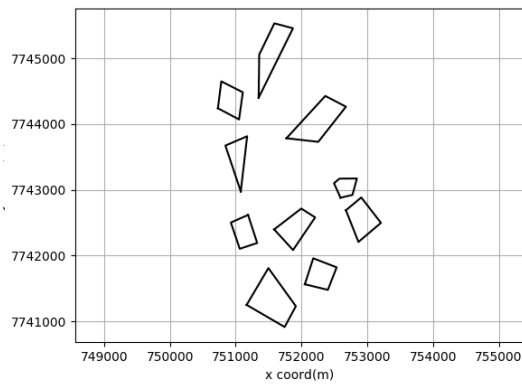
No ultimo cenário de teste, foram escolhido 10 terrenos de formato convexo, como mostrado nas Figuras 43(a) e 43(b).

Com base nas Figuras 43(d), 43(f) e na Tabela 22 nota-se que o método proposto diminuiu tanto o número de curvas quanto a distância total do percurso em relação a rota proposta do *Mission Planner*.

**Figura 43: Fases do processo de comparação de rota entre Mission Planner e método proposto para o cenário 20**



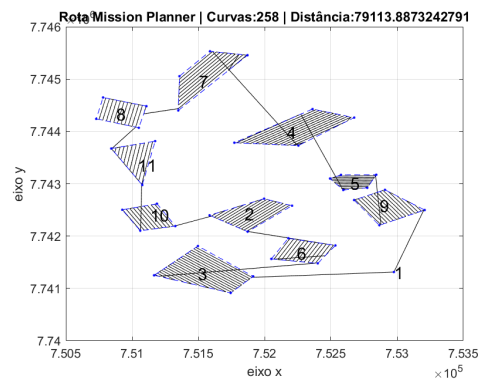
(a) Terreno selecionado no Google Earth Pro



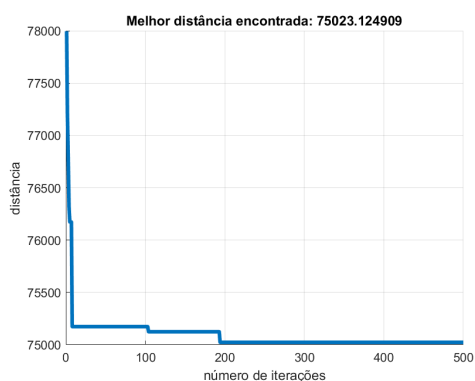
(b) Terreno representado no algoritmo



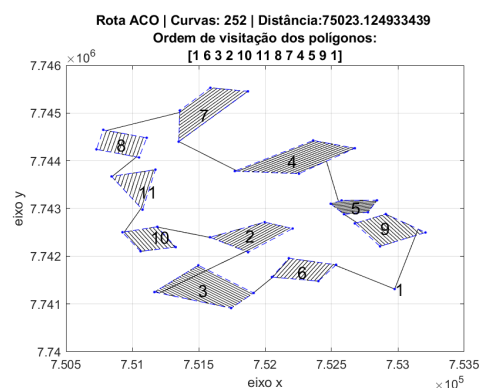
(c) Terreno preenchido com opção Survey no Mission Planner



(d) Rota final do Mission Planner no Matlab



(e) Gráfico de convergência do ACO



(f) Rota final do método proposto no Matlab

Fonte: Própria (2019)

**Tabela 22: Tabela comparativa das rotas obtidas no Mission Planner e no método proposto para o cenário 20**

	Nº de curvas do trajeto	Distância total percorrida	Nº inicial de terrenos	Nº final de terrenos	Ordem de visitação
Rota 1 - Mission Planner	258	79113.88	11	11	1-3-6-2-10-11-8-7-4-5-9-1
Rota 2 - Método proposto	252	75023.12	11	11	1-6-3-2-10-11-8-7-4-5-9-1
Varição percentual	-2.32	-5.17			

Fonte: Própria (2019)

Nestes cenários de teste de 15 a 20 pode-se observar que em cenário maiores, com maior número de terrenos, as diferenças entre as rotas geradas pelos dois métodos aumentam. No geral, usar a decomposição de terreno como regra não trouxe bons resultados, pois em várias situações aumentou o número de curvas da rota.

Nos últimos testes, utilizando cenários de maior extensão e com maior número de terrenos, pode-se notar de uma maneira mais significativa a vantagem da utilização do ACO no algoritmo para tratar as diferentes opções de rotas existentes.

Conforme o número de terrenos aumentou ficou mais difícil determinar visualmente uma rota ótima de visitação de terreno, desta forma o algoritmo de ACO pode ser utilizado como uma ferramenta para conseguir propor rotas de cobertura de áreas otimizadas, que cumpram o objetivo de preencher em completude um dado cenário percorrendo uma distância menor que a proposta apresentada pelo *Mission Planner*, que é largamente utilizado em aplicações de campo.

A parametrização utilizada no algoritmo de ACO permitiu que convergisse para uma solução otimizada, ou parcialmente otimiza em um número reduzido de iterações.

## 5.5 Conclusão

Este trabalho iniciou-se com o intuito de contribuir com a aplicação da tecnologia dos drones na área agrícola, que cresce cada vez mais. O propósito do trabalho foi estudar um ponto que ainda tem sido um limitante para o uso de VANTs em larga escala na área agrícola, contornar a baixa autonomia dos VANTs em frente a grandes áreas de cultivo. Este trabalho explorou a ideia de contribuir para a economia de recursos de energia através que um planejamento otimizado de rota de coberturas de áreas, aplicando a meta-heurística de otimização de colonia de formigas (ACO), juntamente com outros algoritmos computacionais para propor rotas otimizadas, que minimizem o número de curvas e a distância total do percurso.

O fato de aplicar e comparar o método desenvolvido em múltiplos cenários com aspectos diferentes permitiu concluir de uma maneira mais clara que dependendo do cenário do teste uma técnica ou outra embutida no método passa a contribuir de modo mais significativo para a solução, sendo assim, validar um método com base em um único tipo de cenário pode acrescentar um certo viés as conclusões. Como exemplo, com base nos resultados obtidos dos testes, observou-se que adotar a decomposição dos terrenos como regra para diminuir o número de curvas não foi uma boa estratégia de modo geral, pois nota-se que existem casos onde a decomposição excessiva do terreno resulta em muitas subáreas, o que acaba aumentando também o número de curvas do percurso. Inclusive, o que foi levantado na introdução como um problema quando a rota proposta sugere coordenadas fora da área de interesse na solução, pode ser considerada como uma alternativa para a decomposição excessiva, ao invés de sempre decompor os terrenos complexos, caso seja permitido a cobertura fora da área de interesse, esta ação pode apesar de acrescentar na distância total, diminuir o número de curvas do percurso e no final da solução acabar apresentando uma rota que economize mais recursos que uma planejamento restritivo que se limita a área de interesse. A melhor técnica a ser aplicada depende do cenário que está sendo tratado e das restrições do problema.

Por outro lado, diferente do algoritmo de decomposição, uso do algoritmo de ACO não trouxe prejuízo em nenhum cenário de teste, pois nos cenários em que ele não era necessário por conta da existência de poucos terrenos, seu custo computacional foi insignificante já que sua execução foi instantânea. Nos casos dos cenários maiores, o algoritmo apresentou um tempo de execução aceitável e passível de ser usado na prática, já que em média o algoritmo convergiu para rotas otimizadas em menos de 300 iterações. Além disto, a rota apresentada quase sempre era mais curta do que a proposta de rota apresentada pelo *Mission Planner* que é um programa largamente utilizado em tarefas de imageamento aéreo.

Outra técnica que contribuiu de forma positiva para encontrar resultados melhores que o apresentado pelo *Mission Planner* foi o fato de que cada terreno no método proposto possui quatro alternativas de preenchimento, com pontos de entrada e saída diferentes. Em casos de múltiplos terrenos, como no Cenário 13, foi possível observar que mesmo visitando os terrenos na mesma ordem, o método proposto, por possuir quatro alternativas de preenchimento com pontos de entrada e saída diferentes, conseguiu propor uma rota de menor distância em comparação com o método de planejamento do *Mission Planner*, pois este, apesar de possibilitar alterar o ponto de entrada e saída, esta tarefa precisa ser realizada de forma manual em cada um dos terrenos alterando a angulação de preenchimento. O próprio fato de conseguir tratar múltiplas áreas em uma mesma fase de computação entra como contribuição do trabalho, pois no *Mission Planner* para compor rotas que passem por múltiplos terrenos é necessário que os cenários sejam inseridos na ordem em que se deseja visitá-los, já que o *Mission Planner* não realiza qualquer otimização de ordem de visitação.

Este trabalho também buscou contribuir conduzindo as pesquisas realizadas para um método de validação voltado para um cenário mais próximo de uma situação real. Este fator contribuiu para que o método fosse desenvolvido de forma a receber diferentes cenários de entrada e dentro de instantes propor uma rota otimizada em um arquivo similar ao carregados no VANTs para realizar as missões de voo.

Observou-se também que o algoritmo desenvolvido neste trabalho, por se tratar de um método de planejamento otimizado de rota de cobertura que trata múltiplas áreas, com algumas modificações, pode ser utilizado também em outras aplicações. Ainda no cenário agrícola, uma aplicação onde o algoritmo pode obter ótimos resultados é na aplicação de pesticidas com o uso de VANTs. Nestas aplicações o cenário é composto por múltiplos pontos de um terreno que necessitam da aplicação do pesticida, podendo portanto, ser modelado como um cenário de múltiplas áreas a serem visitadas e cobertas com um padrão de preenchimento, muito similar a modelo de aplicação deste trabalho. Outras aplicações agrícolas em que o algoritmo desenvolvido pode ser reaproveitado seria na análise de áreas de mato competição ou áreas com pragas de culturas, onde o VANTs deve visitar essas áreas determinadas para aquisição de imagens detalhadas, aplicações de produtos ou liberação de ovos de predadores naturais.

Este trabalho pôde também contribuir com a literatura fornecendo os dados e cenários de testes para a comunidade. Durante a pesquisa foi observada a falta de cenários de *benchmarks* para algoritmos de CPP, muitos dos trabalhos encontrados pecam na disponibilização dos dados de testes para que seus resultados possam ser reproduzidos. Essa falta de uma base de dados de testes e o conjunto de restrições particulares de cada projeto dificulta uma avaliação



e comparação de soluções entre diferentes trabalhos. Buscando contribuir neste ponto, os cenários de testes, juntamente com as configurações utilizadas neste trabalho estão disponibilizadas no seguinte endereço: [https://github.com/lucasdsfranco/masters\\_research.git](https://github.com/lucasdsfranco/masters_research.git), para que desta forma possa contribuir com outros trabalhos podendo ser utilizado em comparações.

É interessante citar algumas limitações e dificuldades encontradas durante o desenvolvimento deste trabalho. Já de início era de conhecimento que para atingir uma proposta de rota otimizada para cenários com múltiplos terrenos seria preciso tratar um problema do tipo NP-Hard no momento de calcular a otimização de ordem de visitação, porém além deste, um descobriu-se outro problema com esse tipo de complexidade.

A própria decomposição de um polígono também pode ser considerado um problema NP-Hard dependendo do cenário a ser tratado, e que a decomposição de polígono por si só já é um assunto que pode ser explorado como um trabalho em separado, devido as inúmeras técnicas existentes para se decompor um polígono. Para manter o foco no objetivo proposto, foi necessário limitar o escopo da pesquisa e tratar de uma maneira mais superficial a decomposição de polígonos.

Por conta deste trabalho ser multidisciplinar e envolver múltiplos conceitos, foram observados alguns possíveis pontos onde trabalhos futuros poderão atuar. Um dos pontos é a própria área de decomposição de polígonos. Como o modo que foi tratado a decomposição de terrenos não trouxe muitos resultados positivos, trabalhos futuros poderão estudar e identificar as diferentes formas de se decompor polígonos podendo substituir o método apresentado neste trabalho por um outro que seja mais eficiente. Sugere-se também que seja inserido uma rotina adicional no algoritmo, que analise cada caso procurando identificar se decompor o terreno trará benefícios para a solução. Outra alternativa seria uma rotina posterior a decomposição dos terrenos que agregue novamente os sub-terrenos vizinhos que resultaram em orientações de cobertura similares, como é mostrado no trabalho de Yu e Hung (2015) e Yu, Zhou e Zhang (2019).

Outro ponto que pode ser explorado como trabalhos futuros é a modelagem do gasto de energia do drone ao executar um plano de voo, por conta da necessidade de equipamentos para realizar o voo, materiais de medição e até mesmo por conta da complexidade, foram encontrados poucos trabalhos que relacionaram o planejamento de rota com gasto energético.

Independente da área de estudo destes possíveis trabalhos, registra-se aqui a importância de se mapear não só as contribuições positivas da pesquisa, mas também registrar quais técnicas estudadas não surtiram o efeito desejado, de modo a deixar a pesquisa transparente deixando outros pesquisados cientes das dificuldades encontradas durante o trabalho.

## REFERÊNCIAS

ANAC. **Orientações para usuários de Drones**. 2017.

BAGHERIAN, M.; ALOS, A. 3D UAV trajectory planning using evolutionary algorithms: A comparison study. **The Aeronautical Journal**, Cambridge University Press, v. 119, n. 1220, p. 1271–1285, 2015.

BEN-ARIEH, D. *et al.* Transformations of generalized ATSP into ATSP. **Operations Research Letters**, Elsevier, v. 31, n. 5, p. 357–365, 2003.

BLENDER, T. *et al.* Managing a Mobile Agricultural Robot Swarm for a seeding task. In: **IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society**. [S.l.]: IEEE, 2016. p. 6879–6886.

BOCHKAREV, S. **Minimizing Turns in Single and Multi Robot Coverage Path Planning**. Dissertação (Mestrado) — University of Waterloo, 2017.

BOCHKAREV, S.; SMITH, S. On minimizing turns in robot coverage path planning. **2016 IEEE International Conference on Automation Science and Engineering (CASE)**, IEEE, p. 1237–1242, 2016.

BOLLINO, K.; LEWIS, L. Collision-free multi-UAV optimal path planning and cooperative control for tactical applications. In: **AIAA Guidance, Navigation and Control Conference and Exhibit**. [S.l.: s.n.], 2008. p. 7134.

BONYADI, M.; AZGHADI, M.; SHAH-HOSSEINI, H. Population-Based Optimization Algorithms for Solving the Travelling Salesman Problem. **Traveling Salesman Problem**, IntechOpen, n. September 2008, 2008.

BOON, M.; DRIJFHOUT, A.; TESFAMICHAEL, S. Comparison of a Fixed-Wing and Multi-Rotor Uav for Environmental Mapping Applications: A Case Study. **The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences**, Copernicus GmbH, v. 42, p. 47, 2017.

BRASIL. **Principais Conceitos E Expressões Utilizados Na Agricultura De Precisão - AP**. Brasília, DF: Ministério da Agricultura, Pecuária e Abastecimento, 2012.

BULUSEK, B. Coverage Path Planning in Non-Convex Polygon Areas for Orthophotomap Creation Using UAVs. In: **Proc. Conf. IEEE**. [S.l.: s.n.], 2015. v. 111, p. 117.

CEKMEZ, U.; OZSIGINAN, M.; SAHINGOZ, O. Multi colony ant optimization for UAV path planning with obstacle avoidance. **2016 international conference on unmanned aircraft systems (ICUAS)**, IEEE, p. 47–52, 2016.

- COLORNI, A.; DORIGO, M.; MANIEZZO, V. Distributed Optimization by Ant Colonies. In: **Proceedings of the first European conference on artificial life**. Cambridge, MA: [s.n.], 1992. p. 134–142.
- COOMBES, M.; FLETCHER, T.; CHEN, W.; LIU, C. Optimal polygon decomposition for UAV survey coverage path planning in wind. In: **Sensors**. [S.l.]: Multidisciplinary Digital Publishing Institute, 2018. v. 18, n. 7, p. 2132.
- COUTINHO, W.; BATTARRA, M.; FLIEGE, J. The Unmanned Aerial Vehicle Routing and Trajectory Optimisation Problem. **Computers and Industrial Engineering**, Elsevier, v. 120, p. 116–128, 2018.
- DE ALMEIDA, A.; SIMMER, W.; Da Trindade, W.; AMARAL, E. Planejador de Rota baseado na Geometria de Manhattan. **Anais do Computer on the Beac**, p. 379–380, 2013.
- DI FRANCO, C.; BUTTAZZO, G. Coverage Path Planning for UAVs Photogrammetry with Energy and Resolution Constraints. **Journal of Intelligent and Robotic Systems: Theory and Applications**, Springer, v. 83, n. 3-4, p. 445–462, 2016.
- DORIGO, M.; GAMBARDELLA, L. Ant colony system: A cooperative learning approach to the traveling salesman problem. **IEEE Transactions on Evolutionary Computation**, IEEE, v. 1, n. 1, p. 53–66, 1997.
- DORIGO, M.; MANIEZZO, V.; COLORNI, A. The ant system: An autocatalytic optimizing process. In: . [S.l.]: Citeseer, 1991.
- DORIGO, M.; MANIEZZO, V.; COLORNI, A. Ant system: optimization by a colony of cooperating agents. **IEEE Transactions on Systems, man, and cybernetics, Part B: Cybernetics**, v. 26, n. 1, p. 29–41, 1996.
- EISENBEI, H. **UAV Photogrammetry**. Tese (Doutorado) — ETH Zurich, 2009.
- FILIPPIS, L. D.; GUGLIERI, G.; QUAGLIOTTI, F. Path planning strategies for uavs in 3d environments. **Journal of Intelligent & Robotic Systems**, Springer, v. 65, n. 1-4, p. 247–264, 2012.
- GANDOLFO, D.; SALINAS, L.; SERRANO, M.; TOIBERO, J. Energy evaluation of low-level control in UAVs powered by lithium polymer battery. **ISA Transactions**, Elsevier, v. 71, p. 563–572, 2017.
- GLOVER, F.; LAGUNA, M.; MARTI, R. Principles of tabu search. **Approximation algorithms and metaheuristics**, v. 23, p. 1–12, 2007.
- GOLDBARG, E.; GOLDBARG, M.; LUNA, H. **Otimização Combinatória e Meta-Heurísticas: Algoritmos e Aplicações**. [S.l.]: Elsevier Brasil, 2017.
- GONSALVES, T.; SHIOZAKI, T. Solving Capacity Problems as Asymmetric Travelling Salesman Problems. **International Journal of Artificial Intelligence and Applications**, v. 6, n. 2, p. 53–65, 2015.
- GOSS, S.; ARON, S.; DENEUBOURG, J.; PASTEELS, J. Self-organized shortcuts in the Argentine Ant. *Naturwissenschaften*. **Naturwissenschaften**, Springer, v. 76, n. 12, p. 579–581, 1989.

- GRASSÉ, P.; NOIROT, C. Biologie-La Climatisation De La Termitiere Par Ses Habitants Et Le Transport De Leau. **Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences**, Gauthier-Villars/Editions Elsevier, v. 227, n. 17, p. 869–871, 1948.
- GUTIN, G.; PUNNEN, A. **The Traveling Salesman Problem and its Variations**. [S.l.: s.n.], 2006. v. 12.
- HARTMANIS, J. **Computers and intractability: A guide to the theory of NP-completeness**. [S.l.]: Society for Industrial and Applied Mathematics, 1982. v. 24. 90 p.
- HELGAUN, K. **Solving the Equality Generalized TSP using the LKH Algorithm**. [S.l.]: Springer, 2015. 269-287 p.
- HUANG, W. Optimal line-sweep-based decompositions for coverage algorithms. **Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)**, IEEE, v. 1, p. 27–32, 2001.
- INAMASU, R.; BERNARDI, A. **Agricultura de Precisão**. Brasília: Embrapa, 2014. 21-33 p.
- INGERSOLL, B.; INGERSOLL, J.; DEFRANCO, P.; NING, A. UAV Path-Planning using Bezier Curves and a Receding Horizon Approach. In: **AIAA Modeling and Simulation Technologies Conference**. [S.l.: s.n.], 2016. p. 3675.
- JORGE, L.; INAMASU, R. **Uso de veiculos aereos nao tripulados (VANT) em agricultura de precisão**. Brasília: Embrapa, 2014.
- JUN-MAN, K.; YI, Z. Application of an Improved Ant Colony Optimization on Generalized Traveling Salesman Problem. In: **Energy Procedia**. [S.l.]: Elsevier, 2012. v. 17, p. 319–325.
- KEIL, M.; SNOEYINK, J. On the Time Bound for Convex Decomposition of Simple Polygons. **International Journal of Computational Geometry and Applications**, World Scientific, v. 12, n. 03, p. 181–192, 2002.
- KHACHAI, M.; NEZNAKHINA, K. Approximation Schemes for the Generalized Traveling Salesman Problem. **Proceedings of the Steklov Institute of Mathematics**, Springer, v. 299, n. 1, p. 97–105, 2017.
- KHAN, A.; NOREEN, I.; HABIB, Z. On Complete Coverage Path Planning Algorithms for Non-holonomic Mobile Robots: Survey and Challenges. **Journal of Computing and Information Science in Engineering**, v. 33, n. 1, p. 101–121, 2017.
- KHOUI, I.; MINET, P.; ACHIR, N. Unmanned aerial vehicles path planning for area monitoring. In: **2016 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)**. [S.l.]: IEEE, 2016. p. 1–5.
- LAL, R.; SHARDA, A.; PRABHAKAR, P. Optimal multi-robot path planning for pesticide spraying in agricultural fields. In: **2017 IEEE 56th Annual Conference on Decision and Control (CDC)**. [S.l.]: IEEE, 2017. p. 5815–5820.
- LI, Y.; CHEN, H.; ER, M.; WANG, X. Coverage path planning for UAVs based on enhanced exact cellular decomposition method. **Mechatronics**, Elsevier, v. 21, n. 5, p. 876–885, 2011.

- MIAO, X.; LEE, J.; KANG, B. Scalable coverage path planning for cleaning robots using rectangular map decomposition on large environments. **IEEE Access**, IEEE, v. 6, p. 38200–38215, 2018.
- MODARES, J.; GHANEI, F.; MASTRONARDE, N.; DANTU, K. UB-ANC planner: Energy efficient coverage path planning with multiple drones. In: **2017 IEEE International Conference on Robotics and Automation (ICRA)**. [S.l.]: IEEE, 2017. p. 6182–6189.
- NAM, L.; HUANG, L.; LI, X.; XU, J. An approach for coverage path planning for UAVs. In: **2016 IEEE 14th international workshop on advanced motion control (AMC)**. [S.l.]: IEEE, 2016. p. 411–416.
- NOON, C.; BEAN, J. An efficient transformation of the generalized traveling salesman problem. **INFOR: Information Systems and Operational Research**, Taylor and Francis, v. 31, n. 1, p. 39–44, 1993.
- OKSANEN, T.; VISALA, A. Path Planning Algorithms for Agricultural Machines. **Agricultural Engineering International: CIGR Journal**, 2007.
- PEREIRA, J. **Aerial Coverage Path Planning applied to Mapping**. Tese (Doutorado), 2014.
- PHAM, T.; BESTAOUI, Y.; MAMMAR, S. Aerial robot coverage path planning approach with concave obstacles in precision agriculture. In: **2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)**. [S.l.]: IEEE, 2017. p. 43–48.
- PIRES, J.; CHEN, H.; FUHLBRIGGE, T.; LI, X. A review of CAD-based robot path planning for spray painting. **Industrial Robot: An International Journal**, Emerald Group Publishing Limited, 2009.
- PLESSEN, M. Coupling of Crop Assignment and Vehicle Routing for Harvest Planning in Agriculture. **Artificial Intelligence in Agriculture**, Elsevier, v. 2, p. 99–109, 2019.
- QUARITSCH, M. *et al.* Fast aerial image acquisition and mosaicking for emergency response operations by collaborative uavs. In: **Proceedings for the International ISCRAM Conference**. [S.l.: s.n.], 2011. p. 1–5.
- RAO, B.; GOPI, A.; MAIONE, R. The societal impact of commercial drones. **Technology in Society**, Elsevier, v. 45, p. 83–90, 2016.
- SILBERHOLZ, J.; GOLDEN, B. The Generalized Traveling Salesman Problem: A New Genetic Algorithm Approach. In: **Extending the horizons: advances in computing, optimization, and decision technologies**. Boston, MA: Springer, 2007. p. 165–181.
- SINHA, K. **Path Planning for a UAV in an Agricultural Environment to Tour and Cover Multiple Neighborhoods**. Tese (Doutorado), 2017.
- SIVAKUMAR, P.; ELAKIA, K. A Survey of Ant Colony Optimization. **International Journal of Advanced Research in Computer Science and Software Engineering**, v. 6, n. 3, 2016.
- STRIMEL, G.; VELOSO, M. Coverage planning with finite resources. **2014 IEEE/RSJ International Conference on Intelligent Robots and Systems**, IEEE, p. 2950–2956, 2014.

- TORRES, M.; PELTA, D.; VERDEGAY, J.; TORRES, J. Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. **Expert Systems with Applications**, v. 55, p. 441–451, 2016.
- VASQUEZ-GOMEZ, J.; GOMEZ-CASTANEDA, C.; COTE, E.; HERRERA-LOZADA, J. Multicopter UAV coverage planning under wind conditions. In: **2016 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)**. [S.l.]: IEEE, 2016. p. 32–37.
- WANG, P.; ZHANG, Y.; YAN, D. An improved self-adaptive ant colony algorithm based on genetic strategy for the traveling salesman problem. In: **AIP Conference Proceedings**. [S.l.]: AIP Publishing, 2018. v. 1967, n. 1, p. 040046.
- WU, C. *et al.* Energy-efficient coverage path planning for general terrain surfaces. **IEEE Robotics and Automation Letters**, v. 4, n. 3, p. 2584–2591, 2019.
- YAHYANEJAD, S. **Orthorectified mosacking of images from small-scale unmanned aerial vehicles**. Tese (Doutorado) — Alpen-Adria Universität Klagenfurt, 2013.
- YOUSEFIKHOSHBAKHT, M.; DIDEHVAR, F.; RAHMATI, F. Modication of the ant colony optimization for solving the multiple traveling salesman problem. **Romanian Journal of Information Science and Technology**, v. 16, n. 1, p. 65–80, 2013.
- YU, X.; HUNG, J. Coverage path planning based on a multiple sweep line decomposition. In: **IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society**. [S.l.]: IEEE, 2015. p. 4052–4058.
- YU, X.; ZHOU, X.; ZHANG, Y. Collision-Free Trajectory Generation and Tracking for UAVs Using Markov Decision Process in a Cluttered Environment. **Journal of Intelligent and Robotic Systems**, Springer, v. 93, n. 1-2, p. 17–32, 2019.