



**Universidade Federal Rural de Pernambuco**  
**Departamento de Estatística e Informática**  
**Bacharelado em Sistemas de Informação**

## **Projeto FlyFood: Otimização de rotas de drones**

Victor Hugo de Oliveira Gomes

**Recife**

Outubro de 2021

# 1. Introdução

Atualmente os constantes problemas no trânsito, como o grande fluxo de carros e pistas com estruturas precárias, junto ao alto custo do serviço dos entregadores vem dificultando cada vez mais o cumprimento de entregas que necessitam de velocidade, sejam lanchonetes ou restaurantes. E por conta disso surgiu uma nova alternativa, que é a FlyFood, uma empresa que faz essas entregas com o uso de drones.

Os drones são capazes de cumprir a mesma tarefa que o entregador comum, porém, não precisam passar pelo lento e caótico trânsito das cidades, sendo muito mais rápidos que os entregadores convencionais, e não fazem o uso de combustíveis fósseis. Por outro lado, os drones ainda possuem uma relativa baixa capacidade de armazenamento de energia, o que limita o tempo em que esses dispositivos devem efetuar a entrega e ainda retornarem para o estabelecimento de partida.

E esse é o motivo pelo qual se faz necessário o estudo e desenvolvimento de um algoritmo que entregue para o sistema desses drones as rotas mais rápidas possíveis de entrega. E esse documento irá apresentar o processo de desenvolvimento e a eficácia de tal algoritmo.

Para chegar ao objetivo final, antes, se faz necessário a criação de um algoritmo de força bruta para compreender cada necessidade e processo que o código irá precisar satisfazer de forma mais transparente. Passado esse estágio do projeto com sucesso, já se é possível prosseguir para o desenvolvimento de um algoritmo genético.

## 2. Referencial Teórico

O referencial teórico deste trabalho foi organizado em 3 conceitos essenciais para a pesquisa, quais são: Custo computacional, problema do caixeiro viajante e algoritmos genéticos.

### 1. Custo Computacional

Custo computacional ou complexidade computacional trata-se da teoria utilizada para determinar quanto dos recursos do sistema um determinado algoritmo irá consumir. Como desde o sistema operacional até o hardware podem variar muito de um computador para o outro é necessário um método para calcular esses custos que retorne uma medida de custo que sirva para qualquer sistema.

Esses custos podem ser divididos em dois tipos, sendo Custo de Espaço e Custo de Tempo. O primeiro tipo é relativo ao consumo de recursos de

memória, o quanto de memória o algoritmo irá utilizar para armazenar dados e na maioria dos casos se trata da memória volátil do sistema, ou seja, dados temporários para cálculos dentro do próprio algoritmo. O segundo trata de quanto tempo o algoritmo irá levar para efetuar a tarefa que foi entregue e nesse caso é onde o hardware do computador em que o código está sendo usado mais influência.

## 2. Problema do Caixeiro Viajante

O Problema do Caixeiro Viajante, ou Travelling Salesman Problem, é inspirado nas pessoas que trabalhavam vendendo de portas em portas e tinham que viajar entre cidades. Com o objetivo de reduzir custos de recursos como combustível e o tempo que o viajante passaria longe de casa foi criado esse problema que tenta determinar a menor e mais rápida rota a ser seguida para percorrer uma série de pontos.

Tal problema é utilizado na computação para o desenvolvimento de algoritmos de otimização nos quais é procurado o menor custo para a realização de uma tarefa que pode ser até mesmo encontrar a menor rota a ser percorrida e, tal como os vendedores, na computação, ele tem o objetivo de encontrar o menor custo de tempo.

## 3. Algoritmos Genéticos

São algoritmos que usam fundamentos darwinianos e genéticos para achar soluções aproximadas para problemas de otimização e de busca.

Essa classe de algoritmos funcionam, inicialmente, recebendo um conjunto aleatório de possíveis soluções que é chamado de população, essa população passa por processos evolutivos como avaliação, seleção dos que melhor se adaptaram, crossover e mutação para que, por fim, após n ciclos evolutivos, seja retornado a população com o conjunto das soluções que melhor se adaptaram.

## 2.1 Subseção do referencial teórico

O algoritmo que está sendo apresentado neste trabalho se baseou no Problema do Caixeiro Viajante pois tem como objetivo devolver a rota mais rápida que o drone deve percorrer partindo do estabelecimento, efetuando as entregas e retornando ao estabelecimento de partida e para garantir a eficiência do código foi necessário o cálculo do Custo de Tempo e o desenvolvimento da solução em algoritmo genético com o intuito de retornar definitivamente a melhor solução e no tempo mais curto possível de resposta ao sistema.

### 3. Procedimento

Para o desenvolvimento do algoritmo genético foi necessário o desenvolvimento de uma função chamada “PopInicial” que tem a tarefa de retornar uma população aleatória de acordo com o parâmetro que determina o tamanho populacional ou a quantidade de indivíduos, que são listas possuindo, cada uma, uma sequência qualquer de pontos a serem percorridos, ou melhor, uma rota. Após isso vem a criação da classe Fitness que possui a função Distance qual a tarefa é entregar a distância ou custo de cada indivíduo ou rota, também a função “fitnessValue” que deve retornar o valor de fitness da rota e uma outra função chamada “bestFitness” qual retorna uma lista ordenada da rota com o melhor valor de fitness para o pior. Em seguida, através da função “elitismo”, é retornada a lista dos melhores indivíduos da população anterior para que sejam introduzidos para a próxima população/geração.

Os indivíduos resultantes da função “elitismo” são mantidos e, através de uma função chamada “crossOver” são utilizados pares de indivíduos, dos quais, para cada par, são selecionados aleatoriamente dois genes do primeiro indivíduo enquanto os genes restantes para a formação do filho originam do segundo indivíduo e cada um desses filhos é um indivíduo da nova geração. Foi considerado como parâmetro uma taxa de reprodução de 50% para os indivíduos da atual geração.

Foi utilizada como parâmetro para a função “mutação” uma taxa de mutação de 5%, onde o indivíduo que fosse sofrer a mutação teria dois dos seus genes trocados um pelo outro.

Como critério de parada foi considerado 50 iterações.

Por fim uma função chamada “run” vai receber os parâmetros(tamanho da população, taxa de mutação, taxa de reprodução etc), vai criar a primeira população e vai iniciar o loop que irá chamar 50x a função “novaGeração” qual irá criar cada uma das próximas gerações recebendo a anterior como parâmetro e comparando para saber se é a melhor solução.

### 4. Resultados

Explicar quais resultados foram obtidos.

### 5. Conclusão

Relembrar qual era o objetivo do trabalho.

Quais resultados foram alcançados?

Quais conclusões podemos tirar a partir dos resultados?

Quais os trabalhos futuros?

# Referências Bibliográficas

Beraldi, L. C., & Escrivão Filho, E. (2000). Impacto da tecnologia de informação na gestão de pequenas empresas. Revista Ciência da Informação, Brasília, 29(1), 46-50.

Bishop, C. M. (2006). Pattern recognition and machine learning. springer.

BRASIL (2017). Lei Complementar nº 123, de 14 de dezembro de 2006. Institui o Estatuto Nacional da Microempresa e da Empresa de Pequeno Porte. Disponível em: [http://www.planalto.gov.br/ccivil\\_03/leis/LCP/Lcp123.htm](http://www.planalto.gov.br/ccivil_03/leis/LCP/Lcp123.htm). Acesso em 07 dez. 2017

Conselho Regional de Contabilidade do Paraná (2017) . Balanço Patrimonial Disponível em: <https://goo.gl/Lb4jzw>. Acesso em 08 dez. 2017