##### REPORT 3

**Sumário e Primeiro Capítulo Teórico**

##### IDENTIFICAÇÃO

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **NOME** | **e-mail** | **Telefone** |
| **1** | **Luis Felipe dos Santos Gianoni** | **lfelipe2305@hotmail.com.br** | **15981420688** |
| **2** | **Marcos Favoretti Jr** | [**marcos.junior@ethos.ind.br**](mailto:marcos.junior@ethos.ind.br) | **15991642814** |
| **3** | **Patrick Nunes de Souza** | **patrick.n.souza@outlook.com** | **15991459102** |

**TÍTULO:**

Algoritmo de busca em grafos utilizando computação quântica

**LÍDER DO GRUPO:**

Patrick Nunes de Souza

**ORIENTADOR:**

Marcos Fabio Jardini

Data da Entrega: 24 / 04 /2025

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Visto do Orientador

**Sumário**

1. **INTRODUÇÃO**
2. **O CAIXEIRO VIAJANTE**

**2.1 Contextualização do Problema com Grafos**

**2.2 Complexidade NP e P**

**2.3 Soluções da atualidade**

**2. CATEGORIAS DE RECOMENDAÇÃO**

Todo o título deve estar com letras maiúsculas. Texto explicando os tipos que serão abordados no capítulo, a importância dos mesmos e suas vantagens.

**2.1 Baseado em Conteúdo e Filtragem Colaborativa**

Apenas a primeira letra de cada palavra em maiúsculo. Texto sobre o assunto.

2.1.1 Sistemas híbridos

Apenas a primeira letra de cada palavra em maiúsculo e sem negrito. Texto sobre o assunto.

2.1.2 Exemplos de aplicação

Apenas a primeira letra da primeira palavra em maiúsculo e sem negrito. Texto sobre o assunto.

(Utilizar o template do TCC)

**1º CAPÍTULO TEÓRICO (item 2 da Monografia)**

O documento deve conter:

- Lista de Figuras (se for o caso: mais de 5 itens)

- Lista de Tabelas (se for o caso: mais de 5 itens)

- Lista de Abreviaturas e Siglas (se for o caso – ordem alfabética)

- Lista de Símbolos (se for o caso – ordem alfabética)

- SUMÁRIO

- Item 2 da monografia, visto que o item 1 é a INTRODUÇÃO

- Referências

**IMPORTANTE:** Usar fonte Arial, tamanho 12, espaço entrelinhas 1,5, folha A4.

**2 O CAIXEIRO VIAJANTE:**

Nesse projeto será tratado o problema do Caixeiro Viajante, conhecido como “*Travelling Salesman Problem”* (TSP), é um dos problemas mais estudados na área de otimização combinatória e teoria dos grafos. Sua formulação clássica propõe encontrar o caminho mais curto que permite a um caixeiro visitar um conjunto de cidades, passando por cada uma exatamente uma vez e retornando ao ponto de partida. O nome tem origem em uma analogia com os vendedores ambulantes do século XIX, que percorriam diferentes localidades para comercializar seus produtos e, por isso, precisavam planejar rotas eficientes para reduzir tempo e custos.

Segundo Perkins (2021), as origens do Problema do Caixeiro Viajante, ou TSP, remontam ao início do século XIX, mas foi formalizado matematicamente pela primeira vez por um indivíduo chamado Merrill M. Flood em 1930, que tentava resolver um problema de roteamento de ônibus escolares. Posteriormente, adotado por Hassler Whitney em Princeton sob o apelido de "problema dos 48 estados", no qual a rota mais curta era considerada para visitar todos os 48 Estados Unidos contíguos, recebeu finalmente a expressão atual "TSP" em um relatório da RAND Corporation de 1949.

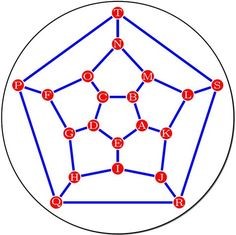
De acordo com Zhou et al. (2019), o TSP tornou-se um paradigma importante na ciência da computação por ser um problema NP-difícil, ou seja, para o qual não se conhece um algoritmo eficiente que resolva todas as instâncias em tempo polinomial, no capítulo 2.3 será abordado mais especificamente sobre problema NP-difícil. Entre as curiosidades que envolvem o TSP, destaca-se o fato de que ele possui aplicações em áreas inesperadas, como na genética (por meio do sequenciamento de DNA), na robótica (planejamento de trajetórias) e até na arte computacional.

2.1 Contextualização do Problema com Grafos

De acordo com Silva, Oliveira e Costa (2022), grafos são estruturas matemáticas fundamentais nas áreas de matemática discreta e estrutura de dados, amplamente utilizadas na ciência da computação para modelar relações e interações entre elementos. Formalmente, um grafo é composto por um conjunto de vértices (ou nós) e por um conjunto de arestas que conectam pares desses vértices. Essa estrutura permite representar, de maneira abstrata, diversos sistemas do mundo real, como redes de transporte, redes de comunicação, redes sociais, entre outros. Por exemplo, em um sistema de transporte, os vértices podem representar cidades e as arestas, as estradas que as conectam. Na área de estrutura de dados, os grafos são essenciais para o desenvolvimento de algoritmos que resolvem problemas como busca em profundidade e largura, detecção de ciclos, caminhos mínimos e otimização de rotas.

Dentro do estudo dos grafos, destaca-se o conceito de ciclo hamiltoniano, que é um caminho fechado que passa por todos os vértices do grafo exatamente uma vez, retornando ao ponto de origem. De acordo com Marcelo de Souza Santos (2016), esse tipo de ciclo foi nomeado em homenagem ao matemático irlandês William Rowan Hamilton, que introduziu a ideia no século XIX por meio de um jogo matemático conhecido como "Icosian Game", como mostrado na figura 1.

Figura 1- tabuleiro do Icosian Game



fonte: disponível em: <https://fr.pinterest.com/pin/465559680219761838>. Acesso em 20 de Abr. 2025.

Segundo Santos, Marcelo de Souza (2016), Esse problema teve origem em 1850 e o adjetivo hamiltoniano foi dado devido ao trabalho do matemático irlandês William R. Hamilton. Hamilton inventou o Icosian Game, um jogo no qual o mundo foi baseado em um grafo dodecaedro. Os vértices desse grafo foram rotulados como cidades e o objetivo do jogo era que um viajante visitasse todas as 20 cidades sem repetir alguma cidade já visitada.

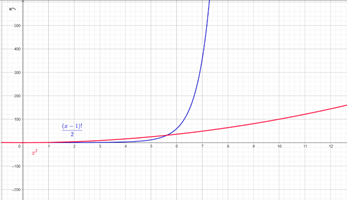
 A existência de um ciclo hamiltoniano em um grafo está diretamente relacionada a diversos problemas práticos, sendo um dos elementos centrais no problema do Caixeiro Viajante.

2.2 Complexidade NP e P

O problema do Caixeiro Viajante está inserido na classe dos chamados problemas NP-difíceis (Non-deterministic Polynomial-time hard), sendo considerado NP-completo em sua versão de avaliação do caminho, onde se busca determinar se existe um percurso com custo inferior ou igual a um valor previamente definido, retornando apenas se é verdadeiro ou falso. A classificação como NP-completo implica que o problema pertence ao conjunto NP (tempo polinomial não determinístico), ou seja, ainda que não exista um algoritmo conhecido que resolva o problema em tempo polinomial para todas as instâncias, qualquer solução proposta pode ser verificada com eficiência, isto é, em tempo polinomial.

De acordo com Perkins (2021), a principal distinção entre problemas de tempo polinomial (polynomial time), como ordenações e buscas simples (com complexidades como O(n), O(n log n), O(n²)), e os de tempo não polinomial (non-polynomial time), como muitos problemas combinatórios, está na escalabilidade da solução. Problemas NP-difíceis, como o Caixeiro Viajante, tendem a apresentar crescimento exponencial ou fatorial no número de possibilidades a serem analisadas por exemplo, com n cidades, existem (n − 1)!/2 rotas possíveis distintas, abaixo na imagem 2 pode-se visualizar escalabilidade da solução entre um problema p e np-dificil. Isso significa que mesmo para instâncias com um número relativamente pequeno de cidades, a quantidade de combinações possíveis pode se tornar computacionalmente inviável de se percorrer por força bruta. Esse crescimento exponencial impõe uma limitação prática: enquanto um algoritmo de tempo polinomial é viável mesmo em grandes volumes de dados, algoritmos de tempo não polinomial tornam-se rapidamente impraticáveis com o aumento da entrada.

Figura 2- Comparação de escabilidade entre NP e P



fonte: autoria própria

2.3 Soluções da atualidade

Diante dessa complexidade, diversos métodos computacionais têm sido empregados para encontrar soluções viáveis ao problema. Os algoritmos exatos, como a técnica de programação dinâmica (ex.: algoritmo de Held-Karp) ou o método de branch and bound, garantem a obtenção da melhor solução possível, mas seu tempo de execução cresce rapidamente à medida que o número de vértices aumenta, o que restringe sua aplicação a instâncias de pequeno ou médio porte.

De acordo com Zhou et al. (2019), Como alternativa, surgem os algoritmos heurísticos e meta-heurísticos, que não garantem a solução ótima, mas produzem respostas satisfatórias em tempo razoável. Dentre esses, destacam-se os algoritmos genéticos, a colônia de formigas, a busca tabu e o simulated annealing, os quais baseiam-se em princípios inspirados na natureza, em processos de otimização probabilística e em estratégias adaptativas para explorar o espaço de soluções. Essas abordagens têm se mostrado especialmente eficazes em cenários reais, como logística, roteamento de veículos, planejamento de produção, redes de distribuição e bioinformática, onde a obtenção de uma solução ótima exata é menos relevante do que alcançar uma solução suficientemente boa dentro de um tempo computacional aceitável.