



BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**CRIAÇÃO DO MÓDULO DE GERAÇÃO DE ROTAS OTIMIZADAS
PARA O MUSEU VIRTUAL TOTAL DE IPORÁ – GO, UTILIZANDO
TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL**

GUILHERME CARVALHO FRANCO

Iporá, GO

2023



INSTITUTO FEDERAL GOIANO - CAMPUS IPORÁ
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**CRIAÇÃO DO MÓDULO DE GERAÇÃO DE ROTAS OTIMIZADAS
PARA O MUSEU VIRTUAL TOTAL DE IPORÁ – GO, UTILIZANDO
TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL**

GUILHERME CARVALHO FRANCO

Trabalho de Conclusão de Curso apresentado ao Instituto Federal Goiano - Campus Iporá, como requisito parcial para a obtenção do Grau de Bacharelado em Ciência da Computação.

Orientadora: Prof. Ma. Luciana Recart Cardoso

Iporá, GO

Dezembro, 2023

Sistema desenvolvido pelo ICMC/USP
Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas - Instituto Federal Goiano

FF825c Franco, Guilherme Carvalho
Criação do módulo de geração de rotas otimizadas para o museu virtual total de Iporá - GO, utilizando técnicas de inteligência artificial / Guilherme Carvalho Franco; orientadora Luciana Recart Cardoso. -- Iporá, 2023.
30 p.

Monografia (Pós-graduação Lato Sensu em em Bacharelado em Ciência da Computação) -- Instituto Federal Goiano, Campus Iporá, 2023.

1. Cultura Cibernética. 2. Inteligencia Artificial. 3. Técnicas de Otimização. I. Cardoso, Luciana Recart, orient. II. Título.



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano, a disponibilizar gratuitamente o documento no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

Identificação da Produção Técnico-Científica (assinale com X)

- Tese
- Dissertação
- Monografia – Especialização
- Artigo - Especialização
- TCC - Graduação
- Artigo Científico
- Capítulo de Livro
- Livro
- Trabalho Apresentado em Evento
- Produção técnica. Qual: _____

Nome completo do autor: Guilherme Carvalho Franco

Curso: Bacharelado em Ciência da Computação

Título do Trabalho: Criação do módulo de geração de rotas otimizadas para o museu virtual total de Iporá – utilizando técnicas de inteligência artificial

Restrições de Acesso ao Documento

Documento confidencial: Não Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: 01/03/2024

O documento está sujeito a registro de patente? Sim Não

O documento pode vir a ser publicado como livro? Sim Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

1. O documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
2. Obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
3. Cumpru quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Iporá, 26 de fevereiro de 2024

Guilherme Carvalho Franco

Assinado eletronicamente pelo o Autor e/ou Detentor dos Direitos Autorais

Ciente e de acordo:

Luciana Recart Cardoso

Assinatura eletrônica do(a) orientador(a)

Documento assinado eletronicamente por:

- Guilherme Carvalho Franco, 2020105231940014 - Discente, em 26/02/2024 17:27:41.
- Luciana Recart Cardoso, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 26/02/2024 17:26:03.

Este documento foi emitido pelo SUAP em 26/02/2024. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 576579
Código de Autenticação: d182f5d6ac



INSTITUTO FEDERAL GOIANO

Campus Iporá

Av. Oeste, Parque União, 350, Parque União, IPORA / GO, CEP 76.200-000

(64) 3674-0400



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

Ata nº 135/2023 - GE-IP/CMPIPR/IFGOIANO

ATA DA SESSÃO DE JULGAMENTO DO TRABALHO DE CURSO DE GUILHERME CARVALHO FRANCO

Aos onze dias, do mês de dezembro de dois mil e vinte e três, às dezesseis horas e quatorze minutos, em sessão pública, a banca examinadora designada na forma regimental pela Coordenação do Curso de Bacharelado em Ciência da Computação, para julgar o trabalho de curso intitulado “**CRIAÇÃO DO MÓDULO DE GERAÇÃO DE ROTAS OTIMIZADAS PARA O MUSEU VIRTUAL TOTAL DE IPORÁ – UTILIZANDO TÉCNICAS DE INTELIGÊNCIA ARTIFICIAL**”, apresentada pelo acadêmico Guilherme Carvalho Franco como parte dos requisitos necessários à obtenção do grau de Bacharel em Ciência da Computação. A banca examinadora foi presidida pela orientadora do trabalho de curso, professora **Ma. Luciana Recart Cardoso**, tendo como membros avaliadores: o professor **Dr. Newarney Torrezão da Costa** e o professor **Dr. Thamer Horbylon Nascimento**. Aberta a sessão, o acadêmico expôs seu trabalho. Em seguida, foi arguido pelos membros da banca e:

(X) tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **aprovação** do acadêmico, sem restrições.

() tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **aprovação** do acadêmico, **condicionada a satisfazer as exigências** listadas na Folha de Modificação de Trabalho de Curso anexa à presente ata, no prazo máximo de 80 (oitenta) dias, a contar da presente data, ficando a professora orientadora responsável por atestar o cumprimento dessas exigências.

() não tendo demonstrado suficiência de conhecimento e capacidade de sistematização do tema de seu trabalho de curso, a banca conclui pela **reprovação** do acadêmico.

Conforme avaliação individual de cada membro da banca, será atribuída a nota nove e meio (9,5) para fins de registro em histórico acadêmico. Os trabalhos foram encerrados às dezesseis horas e cinquenta e um minutos. Nos termos do Regulamento do Trabalho de Curso do Curso de Bacharelado em Ciência da Computação do Instituto Federal Goiano – Campus Iporá, lavrou-se a presente ata que, lida e julgada conforme, segue assinada pelos membros da banca examinadora.

Assinado eletronicamente

Luciana Recart Cardoso, Ma. Orientadora

Newarney Torrezão da Costa, Dr.

Thamer Horbylon Nascimento, Dr.

Documento assinado eletronicamente por:

- Luciana Recart Cardoso, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 11/12/2023 21:14:37.
- Newarney Torrezao da Costa, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 13/12/2023 13:15:01.
- Thamer Horbylon Nascimento, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 14/12/2023 07:43:20.

Este documento foi emitido pelo SUAP em 11/12/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 556173

Código de Autenticação: 201d01856c



INSTITUTO FEDERAL GOIANO

Campus Iporá

Av. Oeste, Parque União, 350, Parque União, IPORA / GO, CEP 76.200-000

(64) 3674-0400

RESUMO

FRANCO, G.C.. **Criação do módulo de geração de rotas otimizadas para o museu virtual total de Iporá – GO, utilizando técnicas de inteligência artificial.** Dezembro, 2023. 30 f. Monografia – (Curso de Bacharelado em Ciência da Computação), Instituto Federal Goiano - Campus Iporá. Iporá, GO.

Este trabalho apresenta o desenvolvimento de um sistema um gerador de rotas otimizadas através do uso de algoritmos bioinspirados e técnicas de inteligência artificial para percorrer todas as localidades históricas informadas no Museu Virtual Total de Iporá — GO. Através da realização de testes de desempenho, eficiência e eficácia, de algoritmos como *Ant Colony Optimization*, *Bat Algorithm*, *Bee Colony Optimization*, *Cuckoo Search*, *Firefly Algorithm*, *Genetic Algorithm* e *Particle Swarm Optimization* para geração de caminhos mínimos, com uma base de dados referente às localidades históricas de Iporá, e aplicação da Fórmula de Haversine para o cálculo da distância geodésica entre os locais especificados. Viabilizou a criação do módulo para gerar rotas otimizadas, através do uso do Ant Colony Optimization em Javascript.

Palavras-chave: Cultura Cibernética, Inteligencia Artificial, Técnicas de Otimização.

LISTA DE FIGURAS

Figura 1 – Pontos históricos no mapa urbano de Iporá.	2
Figura 2 – Fluxograma <i>Ant Colony Optimization</i>	4
Figura 3 – Fluxograma <i>Bat Algorithm</i>	5
Figura 4 – Fluxograma <i>Bee Colony Optimization</i>	6
Figura 5 – Fluxograma <i>Cuckoo Search</i>	7
Figura 6 – Fluxograma <i>Firefly Algorithm</i>	8
Figura 7 – Fluxograma <i>Genetic Algorithm</i>	9
Figura 8 – Fluxograma <i>Particle Swarm Optimization</i>	10
Figura 9 – Fórmula de Haversine	10
Figura 10 – Estrutura da página web	19
Figura 11 – Gráfico de tempo médio de execução	21
Figura 12 – Gráfico de distância média	22
Figura 13 – Modulo de geração de rotas	24
Figura 14 – Geração de rotas	25

LISTA DE TABELAS

Tabela 1 – Localidades, latitude e longitude	12
Tabela 2 – Configurações de <i>hardware</i> utilizadas	13
Tabela 3 – Base de Dados de Entrada - BDE	20
Tabela 4 – Tempo médio obtido em cada algoritmo por máquina	21
Tabela 5 – Distância média obtido em cada algoritmo	22

LISTA DE ALGORITMOS

Algoritmo 1	–	<i>Ant Colony Optimization</i>	13
Algoritmo 2	–	<i>Bat Algorithm</i>	14
Algoritmo 3	–	<i>Bee Colony Optimization</i>	15
Algoritmo 4	–	<i>Cuckoo Search</i>	16
Algoritmo 5	–	<i>Firefly Algorithm</i>	17
Algoritmo 6	–	<i>Genetic Algorithm</i>	18
Algoritmo 7	–	<i>Particle Swarm Optimization</i>	18

SUMÁRIO

1 – INTRODUÇÃO	1
2 – REVISÃO DE LITERATURA	3
2.1 Caminho Mínimo	3
2.2 Algoritmos de Otimização	3
2.3 Algoritmos Bioinspirados	3
2.3.1 <i>Ant Colony Optimization</i>	3
2.3.2 <i>Bat Algorithm</i>	5
2.3.3 <i>Bee Colony Optimization</i>	6
2.3.4 <i>Cuckoo Search</i>	7
2.3.5 <i>Firefly Algorithm</i>	8
2.3.6 <i>Genetic Algorithm</i>	8
2.3.7 <i>Particle Swarm Optimization</i>	9
2.4 Fórmula de Haversine	10
2.5 <i>Smart tourism technologies</i>	11
2.6 <i>Travelling Salesman Problem</i>	11
2.7 Trabalhos Correlatos	11
3 – MATERIAIS E MÉTODOS	12
3.1 Aplicação do <i>Ant Colony Optimization</i>	13
3.2 Aplicação do <i>Bat Algorithm</i>	14
3.3 Aplicação do <i>Bee Colony Optimization</i>	15
3.4 Aplicação do <i>Cuckoo Search</i>	15
3.5 Aplicação do <i>Firefly Algorithm</i>	16
3.6 Aplicação do <i>Genetic Algorithm</i>	17
3.7 Aplicação do <i>Particle Swarm Optimization</i>	17
3.8 Desenvolvimento e Integração	18
4 – RESULTADOS E DISCUSSÕES	20
4.1 Eficiência Temporal	20
4.2 Eficácia dos Resultados	21
4.3 Limitações do Estudo	22
4.4 Escolha do Algoritmo	23
5 – CONSIDERAÇÕES FINAIS	24
5.1 Trabalhos Futuros	26
5.1.1 Ampliação do Escopo	26
5.1.2 Aprimoramento de Hardware	26
5.1.3 Incorporação de Tecnologias Emergentes	27
Referências	28

1 INTRODUÇÃO

A formação de rotas surgiu da necessidade da humanidade de deslocamento e exploração, desde as percorridas por nômades por meio de percursos naturais, como rios, vales e outros, até as geradas pelos seres humanos para trajetos marítimos e aéreos (PAINE, 2014). Através disso nota-se a grande importância das rotas para problemas como, navegação e orientação, turismo e viagens, entre outros.

O turismo, é a oportunidade de explorar novos horizontes e vivenciar novas experiências com culturas distintas. Dentro desse vasto universo, as rotas bem definidas desempenham um papel crucial, guiando os viajantes por caminhos meticulosamente traçados que revelam paisagens deslumbrantes, monumentos históricos e tradições locais únicas (TOWNER; WALL, 1991).

Uma rota bem definida possui como função o suporte à tomadas de decisões e um guia para alcance de um destino específico, portando, tendo como foco a localidade de interesse (SOUFFRIAU; VANSTEENWEGEN, 2010). Além de auxiliar de forma clara e eficiente a direção e orientação para alcance de um objetivo, assumindo um importante papel para o alcance do local desejado.

Assim sendo, com uma presença ampla e diversificada, o emprego de rotas bem definidas pode ser utilizado em diversas áreas e contextos, tais como transporte logístico, navegação e geolocalização, turismo e viagens, exploração e mapeamento, resgate e emergências, dentre outros. Fomentando a necessidade da criação e planejamento de rotas personalizadas e bem definidas (YOO et al., 2017).

O turismo vem acompanhado de diversos benefícios, como a geração de receita local, geração de empregos na região, aumento de visibilidade, estímulo do comércio regional, entre outros fatores positivos (CABUGUEIRA, 2005). Portanto, o turismo traz diversas vantagens econômicas, sociais e culturais para a região, se executado com enfoque na disponibilização de um turismo facilitado (DEVILE, 2009).

Visando o turismo facilitado, uma vertente a ser abordada é a disponibilização de rotas personalizadas e de menor custo, tanto monetário quanto temporal, através da utilização de *smart tourism technology* (STT) (AZIS et al., 2020). Dessa forma, o tempo necessário para o planejamento e estabelecimento de uma rota pode variar dependendo da complexidade abordada (GAVALAS et al., 2017).

À vista disso, rotas personalizadas podem ser criadas através da utilização de algoritmos de otimização, possuindo como parâmetros as localidades de interesse. Uma abordagem para filtrar e selecionar os melhores locais para percorrer é atribuindo critérios de escolha, como custo financeiro, tempo de percurso e distância entre locais (KOCHENDERFER; WHEELER, 2019). Tais critérios, podem influenciar fortemente as tomadas de decisão durante a escolha do percurso.

Semelhante à problemática clássica da computação, denominada *Travelling salesman problem* (TSP), possui o foco em percorrer todas as localidades informadas e retornar ao ponto inicial utilizando a menor distância possível e cada local uma única vez (STASTNÝ; SKORPIL; CIZEK, 2016). Existem inúmeras abordagens algorítmicas para resolução de tal problema, sendo uma delas, por meio de algoritmos de otimização.

Consequentemente, este trabalho possui a proposta de desenvolvimento de um sistema um gerador de rotas otimizadas através do uso de algoritmos bioinspirados e técnicas de inteligência artificial para percorrer todas as localidades históricas informadas no Museu Virtual Total de Iporá — GO. Espera-se como resultados que os habitantes

da região e turistas que saibam da existência de tais localidades e por meio desse portal, consigam encontrá-los presencialmente, de modo fácil e eficaz, por meio da rota sugerida.

Através da utilização de algoritmos como *Ant Colony Optimization* (ACO), *Bat Algorithm* (BA), *Bee Colony Optimization* (BCO), *Cuckoo Search* (CS), *Firefly Algorithm* (FA), *Genetic Algorithm* (GA) e *Particle Swarm Optimization* (PSO), possuindo cada algoritmo a capacidade de se adaptar a demandas específicas e personalizadas. Ou seja, não apenas buscam a menor rota em questão de distância, mas podem também considerar múltiplos critérios de otimização em sua realização, como a economia de combustível, a minimização de emissões de carbono ou a maximização da eficiência operacional (BARBOSA, 2017).

Buscou-se a distância entre dois pontos geográficos mediante a aplicação da Fórmula de Haversine, um método matemático que permite se obter a distância geodésica, considerando a curvatura da Terra, raio de 6371 km (NEVES et al.,). Com base nesses cálculos, é possível obter informações valiosas sobre a proximidade e a relação espacial entre diferentes lugares. Portanto, tal fórmula atua como suporte a criação da base de dados acerca da distância entre os locais históricos em Iporá-GO. A Figura 1, mostra a marcação dos pontos históricos no mapa urbano do município.

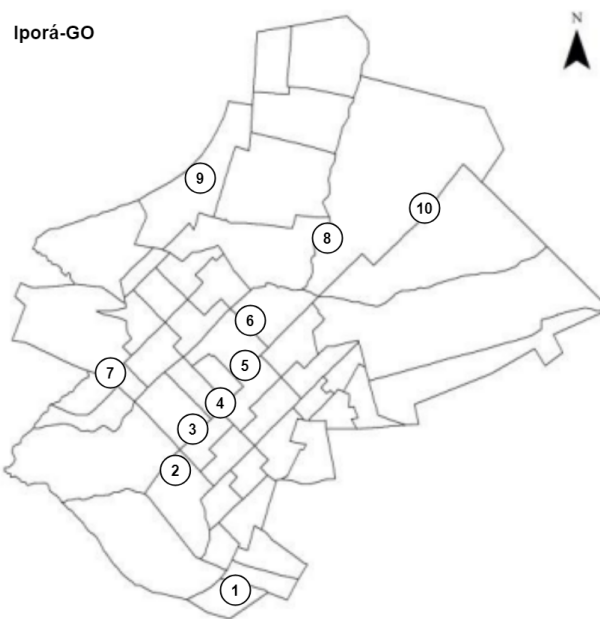


Figura 1 – Pontos históricos no mapa urbano de Iporá.

Fonte: Bueno, 2017.

Concluída a criação da base de dados referente às localidades históricas, foram realizados testes com os algoritmos obtidos durante a revisão da literatura. Os dados obtidos foram usados para cálculo do custo computacional e efetividade dos resultados obtidos, para obtenção do melhor algoritmo e criação do módulo de geração de rotas bem definidas.

Implementado na linguagem *JavaScript* para facilitar a integração com páginas *web*. Essa escolha linguística permite uma conexão fluida com ambientes *web*, oferecendo uma experiência de usuário mais eficiente e intuitiva (ARTZI et al., 2011). A utilização do *JavaScript* proporciona não apenas uma implementação robusta, mas também uma integração mais dinâmica e responsiva, contribuindo para a eficácia do módulo na melhoria da experiência dos usuários ao navegar e explorar o Museu Virtual.

2 REVISÃO DE LITERATURA

2.1 Caminho Mínimo

O caminho mínimo refere-se à rota mais curta entre dois pontos em um determinado contexto. Em diversos campos, como logística, ciência da computação e otimização, o conceito de caminho mínimo é utilizado para encontrar a trajetória mais eficiente ou econômica para alcançar um objetivo específico (ARARUNA, 2013). Essa ideia de percurso mais curto pode ser aplicada em vários contextos, sendo uma abordagem fundamental em diversas áreas como otimizar recursos, tempo ou custos.

De forma geral, uma rota é composta por uma sequência de pontos ou marcos a serem percorridos visando alcançar um destino específico. Esses pontos podem indicar qualquer elemento georreferenciado que facilite a orientação. As rotas podem ser planejadas e otimizadas considerando diferentes critérios, como a menor distância, o tempo de deslocamento mais rápido, a presença de pedágios ou outras preferências particulares do viajante (BARREIRA, 2016).

Uma rota bem definida é caracterizada por um percurso claro estabelecido, acompanhado de instruções precisas para alcançar-se o destino desejado. Tal rota deve caracterizar-se por fácil seguimento, reduzindo a probabilidade de contratempos durante o percurso (FOURNIER, 2022). Além disso, tal rota proporciona a sequência adequada para alcance do destino de maneira eficiente.

2.2 Algoritmos de Otimização

Algoritmos de otimização são modelos matemáticos para resolver problemas complexos, geralmente tal problema possui tempo exponencial de execução. Portanto, emprega-se o uso de algoritmos para obtenção do resultado aceitável em um tempo de execução polinomial, através da análise qualitativa do possível resultado em função do objetivo informado (CORMEN et al., 2022).

Categorizados em diferentes métodos de aplicação, terão cada um sua respectiva abordagem sistemática. A busca local, faz o refinamento iterativo da solução inicial e a busca global, a obtenção pela solução ótima, ou seja, a seleção do melhor resultado entre as possibilidades apontadas (LUGER, 2004).

2.3 Algoritmos Bioinspirados

Algoritmos bioinspirados possuem suas inspirações em fenômenos, eventos ou processos da natureza, através da simulação de mecanismos de evolução, comportamento e comunicação de organismos vivos. Frequentemente utilizados para resolução de problemas de otimização, mediante singularidades e estratégias que cada organismo vivo possui, conseguem eficientemente a resolução de tais problemas (GAO, 2012).

2.3.1 *Ant Colony Optimization*

O *Ant Colony Optimization* é baseado em uma abordagem de otimização estocástica que simula o comportamento de colônias de formigas para encontrar o melhor caminho em um espaço de busca. Nessa abordagem, formigas artificiais constroem soluções para

problemas de otimização, movendo-se pelo espaço de busca com base em informações que indicam caminhos preferidos, cada formiga escolhe seu caminho com base em uma combinação de informações heurísticas e feromônios (BECCENERI et al., 2010). Após várias iterações, os feromônios são atualizados, enfatizando caminhos com melhores soluções. A Figura 2 expõe o fluxograma acerca da sequência de passos que o algoritmo *Ant Colony Optimization* executa conforme ele visa satisfazer o critério de parada informado, variando entre resultado ótimo ou limite de gerações atingido como objetivo do algoritmo.

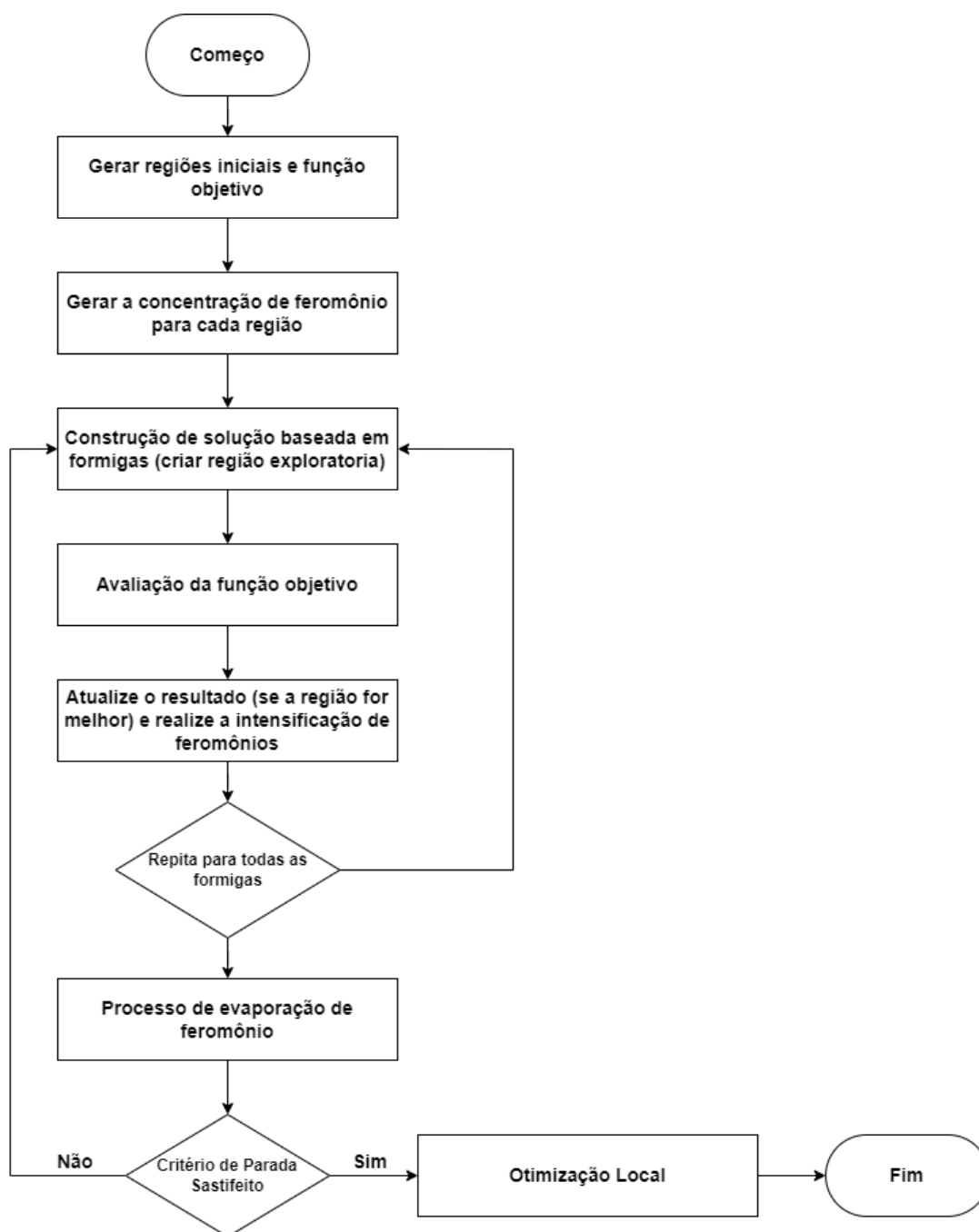


Figura 2 – Fluxograma *Ant Colony Optimization*.
Fonte: Baseado em Barbosa, 2017.

2.3.2 Bat Algorithm

O *Bat Algorithm* é um algoritmo de otimização inspirado no comportamento de morcegos em busca de presas. Começa com um conjunto de morcegos artificiais em posições aleatórias no espaço de busca. Os morcegos emitem “sons”, comportamento de ecolocalização dos morcegos, onde eles emitem pulsos ultrassônicos para detectar a qualidade das soluções próximas e, com base nisso, decidem como e para onde se mover (YANG; HE, 2013). Equilibram a exploração do espaço de busca com a exploração das melhores soluções encontradas, por meio de perturbações aleatórias ou aplicação de regras de distribuição probabilísticas, voos de Lévy, como visualizado na Figura 3. A cada iteração, as posições dos morcegos são atualizadas, visando encontrar uma solução ótima para o objetivo do algoritmo ou atingir o limite de iterações.

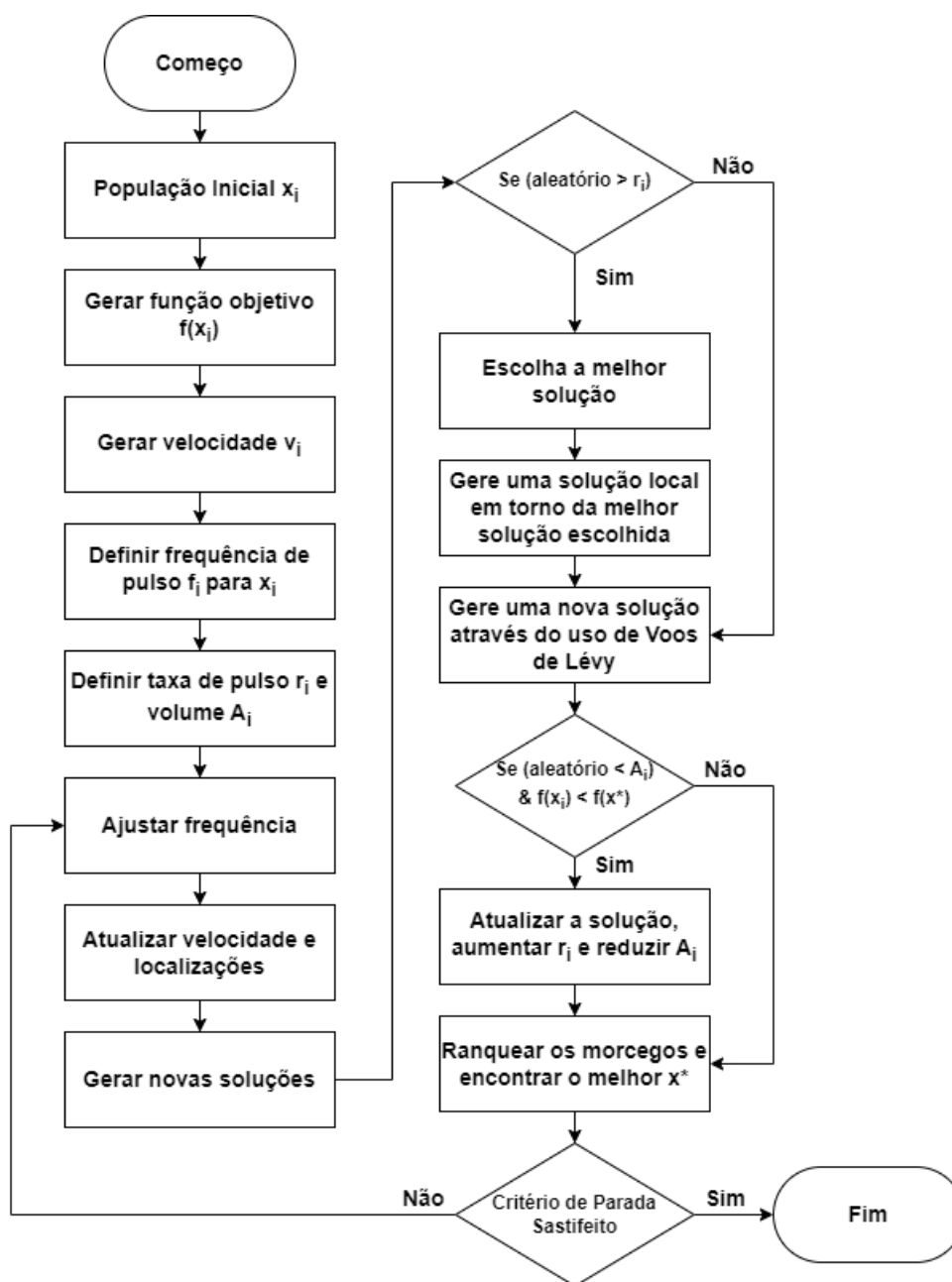


Figura 3 – Fluxograma *Bat Algorithm*.
 Fonte: Baseado em Barbosa, 2017.

2.3.3 *Bee Colony Optimization*

O *Bee Colony Optimization* é um algoritmo de otimização inspirado em como as colônias de abelhas se comportam na natureza. Motivado na forma como as abelhas encontram alimentos, tal algoritmo almeja replicar o processo de como as abelhas colaboram na busca por alimentos, através da avaliação da qualidade da possível solução usando a função objetivo, assim mapeando a solução ótima da trajetória através da simulação de abelhas exploradoras na exploração, busca global, e forrageiras na intensificação, busca local, na busca por plausíveis soluções para um determinado problema (TEODOROVIC et al., 2006), como exemplificado na Figura 4, onde o fluxograma visa gerar a solução ótima ou executar até atingir o limite de iteração.

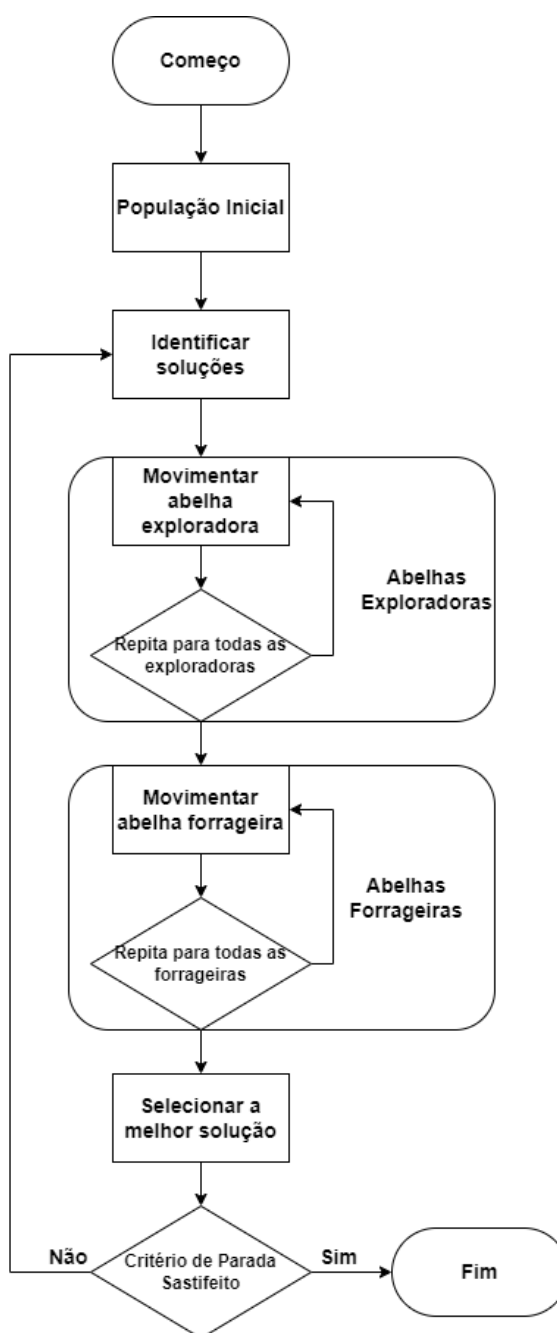


Figura 4 – Fluxograma *Bee Colony Optimization*
Fonte: Baseado em Barbosa, 2017.

2.3.4 Cuckoo Search

O *Cuckoo Search* é um algoritmo de otimização inspirado no comportamento de alguns pássaros “cuckoos” ao depositar seus ovos nos ninhos de outras aves. Utilizado para encontrar soluções otimizadas para problemas complexos, através da simulação do comportamento nativo da espécie. Possuindo a ideia de que os “cuckoos” depositam seus ovos em ninhos de outras aves para cuidarem deles. Da mesma forma, o algoritmo usa populações de soluções candidatas e, ao longo das iterações, algumas soluções são substituídas por outros “cuckoos” que podem ser melhores (GANDOMI; YANG; ALAVI, 2013). Isso ajuda na exploração do espaço de busca para encontrar soluções ótimas, como observado na Figura 5.

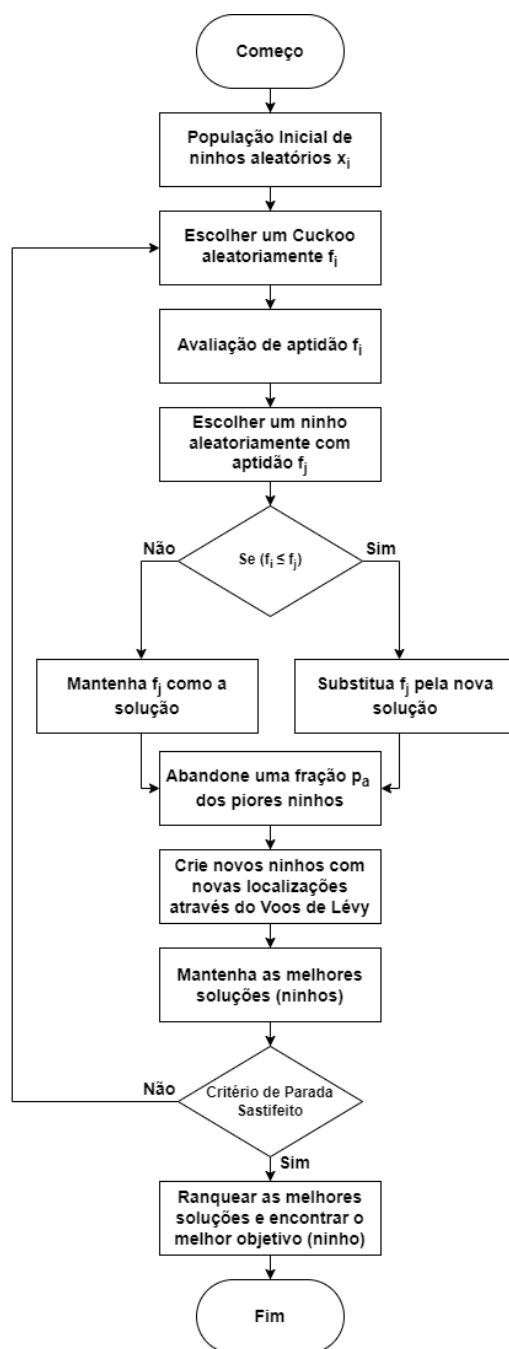


Figura 5 – Fluxograma *Cuckoo Search*.
Fonte: Baseado em Barbosa, 2017.

2.3.5 *Firefly Algorithm*

O *Firefly Algorithm* é um algoritmo de otimização inspirado no comportamento de vaga-lumes na natureza. Esse algoritmo é utilizado para resolver problemas de otimização, especialmente problemas nos quais se busca encontrar o mínimo global de uma função. A ideia principal é modelar o comportamento de vaga-lumes, que emitem luz para atrair parceiros (FISTER et al., 2013). No contexto do algoritmo, as soluções candidatas são representadas como vaga-lumes, e a intensidade da luz emitida por um vaga-lume é associada à qualidade da solução. O objetivo é que os vaga-lumes se movam no espaço de busca de soluções, atraindo-se uns aos outros com base na intensidade da luz, de modo a convergir para uma solução ótima, como proposto na Figura 6.

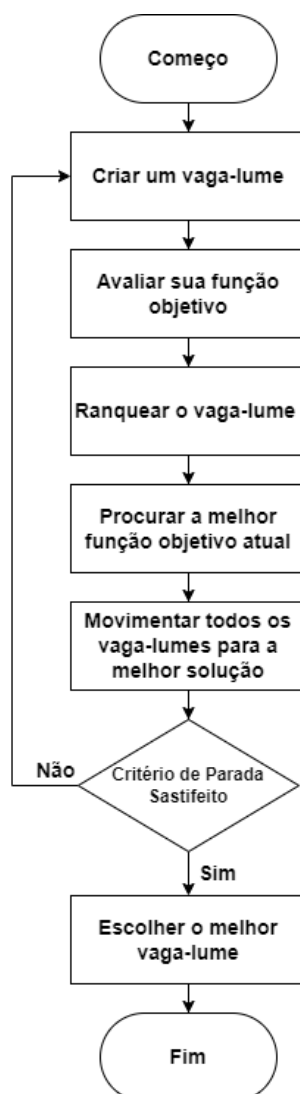


Figura 6 – Fluxograma *Firefly Algorithm*.

Fonte: Baseado em Barbosa, 2017.

2.3.6 *Genetic Algorithm*

O *Genetic Algorithm* é uma técnica de otimização que se inspira na seleção natural e genética para encontrar soluções aproximadas ou ótimas para problemas complexos. Opera com uma população de soluções, avalia sua aptidão, realiza as operações genéticas,

seleção, cruzamento e mutação, e repete o processo ao longo de várias gerações até o critério de parada satisfeito ou valor máximo de gerações ser atingido (PACHECO et al., 1999), como visualizado na Figura 7, assim convergindo para uma solução mais adequada ao longo de sua execução.

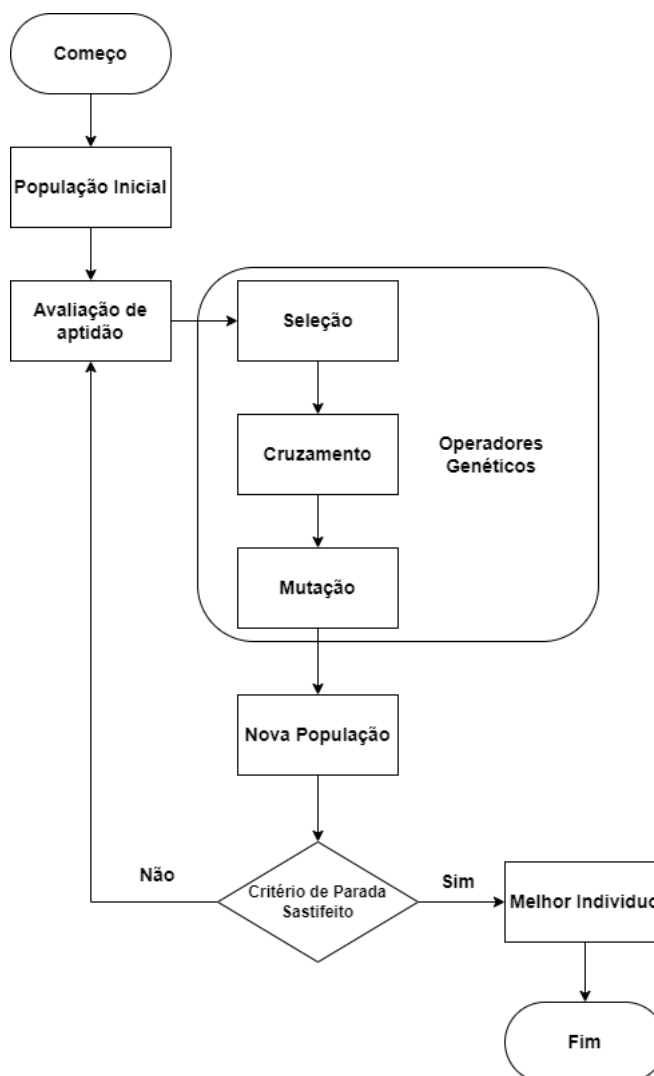


Figura 7 – Fluxograma *Genetic Algorithm*.

Fonte: Baseado em Barbosa, 2017.

2.3.7 *Particle Swarm Optimization*

O *Particle Swarm Optimization*, é uma técnica de otimização computacional que se inspira no comportamento social observado em pássaros, insetos, peixes e colônias. Nessa abordagem, os membros do grupo colaboram de maneira cooperativa na busca pela solução mais eficiente para um determinado problema (KENNEDY; EBERHART, 1995). O algoritmo tem como finalidade a otimização de uma função objetivo, que reflete a métrica de desempenho a ser maximizada ou minimizada, como observado na Figura 8. As partículas deslocam-se pelo espaço de busca, ajustando suas posições e qualidades conforme sua função objetivo.

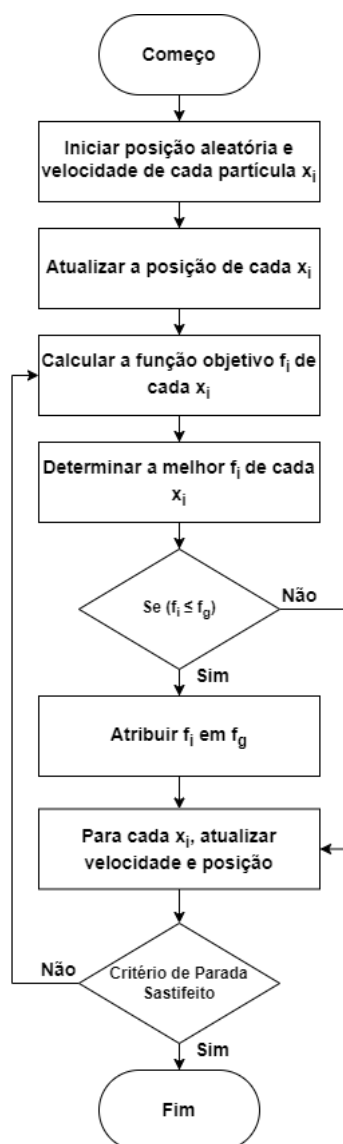


Figura 8 – Fluxograma *Particle Swarm Optimization*.
Fonte: Baseado em Barbosa, 2017.

2.4 Fórmula de Haversine

Fórmula de Haversine é um método matemático para cálculo da distância entre dois pontos em uma esfera, como visualizado na Figura 9, comumente utilizada para o cálculo da distância geográfica de duas localidades sobre a curvatura da Terra. Através da latitude e longitude respectivas de cada localidade e a curvatura aproximada da Terra, é possível adquirir um resultado aproximado da distância, em quilômetros ou milhas, entre dois locais informados (SINNOTT, 1984).

$$d = 2r \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Figura 9 – Fórmula de Haversine
Fonte: Sinnott, 1984.

2.5 *Smart tourism technologies*

As *Smart Tourism Technologies* são empregadas para aprimorar a experiência dos turistas e aumentar a eficiência das atividades turísticas. Essas tecnologias visam facilitar e melhorar diferentes aspectos da indústria do turismo, como a disponibilidade de informações relevantes, a interação com o ambiente e a personalização das rotas, os quais são fatores-chave que influenciam a experiência do usuário (JEONG; SHIN, 2020).

Em suma, as *smart tourism technologies* visam aprimorar a experiência dos turistas. Isso é alcançado por meio do uso de avanços tecnológicos, como inteligência artificial, análise de dados, internet das coisas, dentre outros. Tais avanços possibilitam a obtenção de informações relevantes, agrupam os melhores resultados e personalizam as experiências para melhoria do conforto e da facilidade dos usuários (BAGGIO; MICERA; CHIAPPA, 2020).

2.6 *Travelling Salesman Problem*

O *Travelling salesman problem* é um desafio de otimização, categorizado como resolvível em tempo exponencial. Seu foco é estipular a rota mais curta percorrível, visitando cada local designado exatamente uma única vez e retornando ao local de origem. Tendo sua aplicabilidade em diversas áreas, como logística, distribuição, circuitos eletrônicos e turismo, dentre outras (JÜNGER; REINELT; RINALDI, 1995).

2.7 *Trabalhos Correlatos*

Segundo Xiujuan, em seu artigo *Intelligent Tourism Route Optimization Method based on the Improved Genetic Algorithm*, descreve como a combinação do *immune algorithm* e do *genetic algorithm* pode melhorar os resultados no *Travelling Salesman Problem* (MA, 2016), com foco na verificação da eficácia do algoritmo proposto em um sistema de recomendação de rotas inteligentes, através da representação da distância entre cada lugar em uma matriz de peso.

De acordo com Wenqiang, em seu trabalho *Multiobjective Particle Swarm Optimization with Improved Selection Strategy for Route Optimization*, propõe para o problema de geração de rotas multiobjetivos a aplicação de um algoritmo denominado *improved selection strategy based multiobjective particle swarm optimization* para redução do tempo de execução e melhoria da qualidade de resultado gerado (ZHANG et al., 2019).

Conforme Mayank, em sua pesquisa *Nature Inspired Route Optimization in Vehicular Adhoc Network* (BHATT et al., 2016), visa elencar rotas otimizadas para chegar ao destino, evitando colisões de tráfego, usando uma abordagem heurística de otimização, dividido em previsão da localização destino, a formação de uma região e seleção da rota otimizada, através do uso do *bat algorithm*.

Segundo Jiang, em seu artigo *Optimization Model of Tourist Transportation Route based on Multi-Neuron Algorithm*, utiliza-se do *multi-neural algorithm* para cálculo e pontuação de rotas pré-definidas e recomendação (XIONG, 2022), a fim de promover a recomendação da melhor rota, tanto em questão de visualização dos melhores lugares quanto transporte adequado.

3 MATERIAIS E MÉTODOS

Primordialmente foi realizada uma pesquisa exploratória para construção do material bibliográfico de apoio a respeito de algoritmos de bioinspirados e de otimização, através do escopo literário de geração de rotas otimizadas. Realizado tal pesquisa em fontes secundárias como repositórios de literatura científica, IEEE Xplore, SBCOpenLib (SOL) e CAFe CAPES.

Foram adquiridos os algoritmos a serem utilizados para resolução de problemas de otimização na geração de rotas bem definidas, família de algoritmos apresentado por Barbosa em sua dissertação “Algoritmos bio-inspirados para solução de problemas de otimização” (2017). Escolhidos para realização dos testes de custo computacional e efetividade de resultados o *Ant Colony Optimization* (ACO), *Bat Algorithm* (BA), *Bee Colony Optimization* (BCO), *Cuckoo Search* (CS), *Firefly Algorithm* (FA), *Genetic Algorithm* (GA) e *Particle Swarm Optimization* (PSO).

Para execução dos testes foi criada uma base de dados com as latitudes e longitudes, como visualizado na Tabela 1, de locais históricos na cidade de Iporá-GO. Para a aplicação da Fórmula de Haversine, cálculo da distância geodésica entre os locais especificados, foi utilizado como base o valor de 6371 km para o raio da Terra, assim gerando a Base de Dados de Entrada (BDE) a ser utilizada pelos algoritmos.

Tabela 1 – Localidades, latitude e longitude

Local	Latitude	Longitude
1	-16.459647	-51.115563
2	-16.447236	-51.121254
3	-16.443205	-51.120531
4	-16.441547	-51.119248
5	-16.437327	-51.115416
6	-16.432358	-51.116129
7	-16.438913	-51.128335
8	-16.427266	-51.109344
9	-16.425465	-51.121638
10	-16.421133	-51.097933

Durante a execução da família de algoritmos foi escolhida uma abordagem quantitativa para catalogação e análise dos resultados, em 3 máquinas diferentes especificadas na Tabela 2. A seleção desta abordagem visa quantificar de maneira objetiva e mensurável os dados obtidos, permitindo uma análise robusta e a identificação de padrões ou disparidades significativas entre os desempenhos dos algoritmos selecionados. Para que posteriormente fosse realizado a eleição do melhor algoritmo, a abordagem quantitativa proporcionará uma base sólida para a comparação das métricas de desempenho. Além de proporcionar uma análise objetiva dos resultados considerados, fatores como eficiência e tempo de processamento.

A escolha do algoritmo mais eficaz é crucial para otimizar o desempenho do sistema de geração de rotas otimizadas e, conseqüentemente, aprimorar a qualidade geral da interação humano-máquina. A eficiência do algoritmo não apenas impacta diretamente

na rapidez e na precisão da elaboração das rotas, mas também influencia na experiência do usuário, tornando a interação mais eficaz, intuitiva e adaptada às escolhas específicas do usuário.

Tabela 2 – Configurações de *hardware* utilizadas

Máquina	Processador	Memória RAM	Armazenamento	Placa de Vídeo	Sistema Operacional
1	Intel(R) i5-2310	8GB DDR3	512GB SSD	Intel(R) HD Graphics	Win 10
2	Intel(R) i5-6300HQ	8GB DDR3	1TB HD	NVIDIA GTX 960M	Win 10
3	Intel(R) i5-8265U	8GB DDR4	240G SSD	Intel(R) UHD 620	Win 10

3.1 Aplicação do *Ant Colony Optimization*

Primeiramente foi realizada a execução do *Ant Colony Optimization* com 10.000 iterações do algoritmo, possuindo um tamanho de população de 5 formigas (N) e com o fator de evaporação de feromônio de 0.5 (q) e com limite de 20 repetições (M). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 1 exemplifica o pseudocódigo do *Ant Colony Optimization* utilizado. Aonde o algoritmo recebe o BDE como entrada de dados da distância entre os locais. Logo após é definida a função objetivo do algoritmo, função responsáveis pela análise da qualidade das soluções obtidas durante a execução do código, além de definir e gerar a população inicial de formigas. Posteriormente, são realizadas as construções e avaliações das soluções, para por fim, realizar o processo do ciclo de feromônio até o critério de parada ser atingido.

Algoritmo 1: *Ant Colony Optimization*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Gerar população inicial de formigas $x_i (i = 1, 2, \dots, N)$;
- 5 Avaliar as formigas da população inicial usando a função objetivo;
- 6 **repita**
- 7 $j \leftarrow 0$;
- 8 **enquanto** $j < N$ **faça**
- 9 Construir novas soluções;
- 10 Avaliar novas soluções usando função objetivo;
- 11 $j \leftarrow j + 1$;
- 12 Evaporar feromônio;
- 13 Ordenar soluções;
- 14 Atualizar feromônio;
- 15 **até** critério de parada ser atingido;

3.2 Aplicação do *Bat Algorithm*

O segundo algoritmo executado foi o *Bat Algorithm* com 10.000 iterações do algoritmo, com o tamanho da população de 10 morcegos (N), taxa de pulso de 0.2 (r_0) e frequência de pulso de 0.5 (f_0) e com limite de 50 repetições (M). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 2 exemplifica o pseudocódigo do *Bat Algorithm* utilizado. Aonde o algoritmo recebe o BDE como entrada de dados da distância entre os locais, logo após é definida a função objetivo do algoritmo, além de gerar a população inicial de morcegos e definir a taxa de emissão e a frequência inicial de pulsos. A partir de então, realiza a geração de novas possíveis soluções e atualização da transição de localidade dos morcegos. Por via de escolha de um valor randômico é gerado uma nova solução ao entorno da melhor solução obtida até o momento, caso não seja gerada ao entorno aleatório das localidades já obtidas até o momento. Para mais adiante verificar se o valor randômico é menor que a média de amplitude sonora (A_i) ou melhor, que o resultado da função objetivo da melhor possibilidade de solução existente para assim ser atualizado o melhor morcego e os valores correspondentes a taxa de pulso e media de amplitude. Por fim é efetuada a análise das diversas soluções disponíveis que são categorizadas para selecionar a opção mais apropriada conforme a função objetivo.

Algoritmo 2: *Bat Algorithm*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Gerar população inicial de N morcegos com posição x_i e velocidade v_i para todo ($i = 1, 2, \dots, N$);
- 5 Definir a taxa inicial de emissão de pulsos r_0 e a frequência inicial f_0 ;
- 6 Avaliar os morcegos da população inicial usando a função objetivo;
- 7 **repita**
- 8 Gerar novas soluções;
- 9 Ajustar frequência;
- 10 Atualizar a velocidade e a posição dos morcegos;
- 11 **para** $i = 1$ até número de indivíduos **faça**
- 12 **se** número randômico $< r_i$ **então**
- 13 Selecionar uma solução entre as melhores soluções;
- 14 Gerar uma solução local ao redor da melhor solução;
- 15 Gerar nova solução através de um voo randômico;
- 16 **se** número randômico $< A_i$ ou $f(x_{melhor})$ **então**
- 17 Aceitar nova solução;
- 18 Aumentar r_i e reduzir A_i ;
- 19 Avaliar as posições dos morcegos;
- 20 Ranquear as posições dos morcegos;
- 21 Encontrar a melhor posição atual para os morcegos x_{melhor} ;
- 22 **até** critério de parada ser atingido;

3.3 Aplicação do *Bee Colony Optimization*

Em sequência, foi realizada a execução do *Bee Colony Optimization* com 10.000 iterações do algoritmo, com o tamanho da população de 10 fontes de alimento (N), quantidade de 10 abelhas exploradoras, quantidade de 1 abelha forrageira e limite de 100 tentativas (M). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 3 demonstra o pseudocódigo do *Bee Colony Optimization* utilizado. Aonde o algoritmo recebe o BDE como entrada de dados da distância entre os locais, logo após é definida a função objetivo do algoritmo e o número de tentativas antes do abandono da fonte de alimento. Esta é abandonada pela abelha trabalhadora correspondente, após tais definições são avaliadas as fontes de alimentos através da função objetivo declarada. A próxima etapa é posicionar as abelhas exploradoras em fontes de alimentos aleatórios e realizar o cálculo da probabilidade de escolha de cada fonte de alimento pela abelha forrageira, a fim de para gravar o resultado mais eficaz até o critério de para ser satisfeito.

Algoritmo 3: *Bee Colony Optimization*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Definir o valor de abandono da fonte de alimento;
- 5 Gerar população inicial de fontes de alimentos $x_i (i = 1, 2, \dots, N)$;
- 6 Avaliar as fontes de alimento da população inicial usando a função objetivo;
- 7 **repita**
- 8 Posicione as abelhas exploradoras em suas fontes de alimento;
- 9 Calcule a probabilidade de escolha da fonte de alimento pelas abelhas forrageiras;
- 10 Posicione as abelhas forrageiras nas fontes de alimento dependendo de seus valores de probabilidade;
- 11 Interrompa a exploração de fontes com limite de tentativas esgotado;
- 12 Envie abelhas exploradoras para buscar novas fontes de alimento;
- 13 Memorize a melhor fonte de alimento encontrada até então;
- 14 **até critério de parada ser atingido;**

3.4 Aplicação do *Cuckoo Search*

Posteriormente, foi realizada a execução do *Cuckoo Search* com 10.000 iterações do algoritmo, com o tamanho da população de 10 (N) e probabilidade de abandono de 0.3 (p_a) e limite de 100 tentativas (M). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 4 exemplifica o pseudocódigo do *Cuckoo Search* utilizado. Aonde o algoritmo recebe o BDE como entrada de dados da distância entre os locais, logo após é definida a função objetivo do algoritmo e a probabilidade de abandono de um ninho. Após gerar e avaliar, utilizando a função objetivo, os ninhos hospedeiros, aleatoriamente, um cuco é selecionado e sua qualidade de do seu ninho é comparada com a de outro ninho escolhido ao acaso. Esse procedimento permite a introdução de novas soluções potencialmente melhores, inspiradas pelo comportamento do cuco na natureza,

reforçando a capacidade do algoritmo de explorar de forma abrangente o espaço de busca em visando encontrar as soluções mais promissoras. A partir disso então substituir os piores ninhos devido à probabilidade de abandono e substituindo-os por novos objetivando a escolha dos melhores ninhos até o critério de parada ser satisfeito.

Algoritmo 4: *Cuckoo Search*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Definir a probabilidade de p_a ;
- 5 Gerar população inicial de ninhos hospedeiros $x_i (i = 1, 2, \dots, N)$;
- 6 Avaliar ninhos da população inicial usando a função objetivo;
- 7 **repita**
- 8 Obter um cuco aleatoriamente através do voos de Lévy;;
- 9 Avaliar a qualidade do ninho através de f_i ;
- 10 Escolher aleatoriamente um novo ninho j ;
- 11 **se** $f_i > f_j$ **então**
- 12 └ Substitua o ninho i pelo novo ninho j ;
- 13 Uma fração p_a dos piores ninhos são abandonados e novos ninhos são
 construídos;
- 14 Manter os melhores ninhos;
- 15 Ordenar os ninhos;
- 16 Escolher o melhor ninho;
- 17 **até critério de parada ser atingido;**

3.5 Aplicação do *Firefly Algorithm*

Em continuidade, foi realizada a execução do *Firefly Algorithm* com 10.000 iterações do algoritmo, com o tamanho da população de 10 (N) vaga-lumes e limite de 100 tentativas (M). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 5 exemplifica o pseudocódigo do *Firefly Algorithm* utilizado. Através da utilização do BDE como entrada de dados da distância entre os locais propostos, foi estabelecida a função objetivo e a população inicial, que por meio desta é realizada a avaliação dos vaga-lumes gerados com base na função objetivo, para então realizar a escolha mais atrativa do vaga-lume com maior brilho. Além disso, essa relação direta entre a atratividade dos vaga-lumes e seu brilho, bem como a redução simultânea de ambos com o aumento da distância entre eles, desempenha um papel fundamental no estabelecimento das interações dinâmicas no contexto do algoritmo de vaga-lumes. Por meio desta atratividade realizar o movimento dos vaga-lumes em direção ao vaga-lume mais atrativo, ou brilhante. Caso não exista, ou seja possível realizar tal movimentação em relação à atratividade, tem-se que mover o vaga-lume aleatoriamente no espaço de busca até que o critério de parada seja satisfeito.

Algoritmo 5: Firefly Algorithm

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Gerar população inicial de vaga-lumes $x_i (i = 1, 2, \dots, N)$;
- 5 Definir o coeficiente de absorção de luz;
- 6 Avaliar os vaga-lumes da população inicial usando a função objetivo;
- 7 **repita**
- 8 **para cada vaga-lume i faça**
- 9 **para cada vaga-lume j faça**
- 10 **se $l_i < l_j$ então**
- 11 └ Mover vaga-lume i em direção ao vaga-lume j ;
- 12 └ Variar a atratividade com distância;
- 13 └ Calcular novas soluções e atualizar a intensidade de luz;
- 14 Ordenar os vaga-lumes e encontrar a melhor solução global;
- 15 **até critério de parada ser atingido;**

3.6 Aplicação do Genetic Algorithm

Em continuidade, foi realizada a execução do *Genetic Algorithm* com 10.000 iterações do algoritmo, com o tamanho da população de 500 cromossomos (N), taxa de 0.1 de mutação e máximo de 100 gerações (M). Possui complexidade geralmente expressa por $O(N \times M \times F)$, notação Big O, onde F representa a complexidade dos operadores genéticos. O Algoritmo 6 exemplifica o pseudocódigo do *Genetic Algorithm* utilizado. Ao empregar o BDE como entrada de dados para representar as distâncias entre os locais propostos, para então definir a função objetivo, a qual serve como critério fundamental para realizar o processo de avaliação dos cromossomos gerados. Após essa avaliação, torna-se viável a aplicação dos operadores genéticos, seleção, cruzamento e mutação, visando à perpetuação dos cromossomos e aprimoramento dos resultados potenciais. É perpetuada a transferência de características promissoras entre cromossomos e a mutação introduz variações exploratórias no espaço de busca, a fim de encontrar o melhor cromossomo até o critério de parada ser atingido.

3.7 Aplicação do Particle Swarm Optimization

Em seguida foi realizada a execução do *Particle Swarm Optimization* com 10.000 iterações do algoritmo, com o tamanho da população de 20 indivíduos (N), máximo de 100 gerações (M), peso de 0.5 de inércia (w), peso de 2.0 cognitivo (c_1) e peso de 2.0 social (c_2). Possui complexidade geralmente expressa por $O(N \times M)$, notação Big O. O Algoritmo 7 exemplifica o pseudocódigo do *Particle Swarm Optimization* utilizado. Ao empregar o BDE como entrada de dados para representar as distâncias entre os locais propostos, para então definir a função objetivo, peso de inércia, coeficiente cognitivo e social. Para então por meio da contínua troca de informações entre os indivíduos, representados como partículas em uma população, ou enxame. Essa constante interação informativa, no âmbito da otimização, desempenha um papel crucial na exploração eficiente do espaço de busca,

Algoritmo 6: *Genetic Algorithm*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de cromossomos;
- 4 Definir taxa de cruzamento e taxa de mutação;
- 5 Definir número máximo de gerações;
- 6 Gerar população inicial de cromossomos $x_i (i = 1, 2, \dots, N)$;
- 7 Avaliar os cromossomos da população inicial usando a função objetivo;
- 8 **repita**
- 9 Selecionar os cromossomos para reprodução;
- 10 Aplicar os operadores de cruzamento;
- 11 Aplicar os operadores de mutação;
- 12 Avaliar os cromossomos da população atual usando a função objetivo;
- 13 Manter os melhores cromossomos;
- 14 **até critério de parada ser atingido;**

capacitando a população a evoluir dinamicamente em direção a soluções cada vez mais promissoras. Dessa forma, melhorando o resultado no âmbito do espaço de busca, até que o critério de parada seja atingido.

Algoritmo 7: *Particle Swarm Optimization*

Data: BDE

- 1 **início**
- 2 Definir a função objetivo $f(x)$;
- 3 Definir o número de indivíduos;
- 4 Definir número máximo de gerações;
- 5 Gerar população inicial de partículas com posição x_i e velocidade v_i para todo $(i = 1, 2, \dots, N)$;
- 6 Definir o peso de inércia w ;
- 7 Definir os coeficientes cognitivo (c_1) e social (c_2);
- 8 Avaliar os indivíduos da população inicial usando a função objetivo;
- 9 Defina p_{best} como a melhor posição da partícula i ;
- 10 Defina g_{best} como a melhor solução encontrada de todas as partículas;
- 11 **repita**
- 12 Calcule a nova posição utilizando a função objetivo;
- 13 Atualize a velocidade de cada partícula i ;
- 14 Atualize a melhor posição de cada partícula i ;
- 15 Atualize a melhor posição encontrada por todas as partículas;
- 16 **até critério de parada ser atingido;**

3.8 Desenvolvimento e Integração

Considerando os resultados obtidos durante os testes de eficiência e eficácia dos resultados apresentados pelos algoritmos, que serão melhor detalhados na seção seguinte,

após a execução da comparação dos resultados para eleger o melhor desempenho algorítmico entre as opções selecionadas, essa avaliação criteriosa proporcionará percepções valiosas para a escolha do algoritmo mais adequado às necessidades específicas do contexto. Realizada a escolha do algoritmo, foi iniciado o desenvolvimento do sistema de geração de rotas, a partir das localidades informadas no Museu Virtual Total de Iporá-GO e do melhor algoritmo escolhido.

Concluído o desenvolvimento do módulo de geração de rotas foi desenvolvida uma página *web*, para referência ao sistema no Museu Virtual Total de Iporá-GO, como visualizado na estrutura da Figura 10. Para contribuir com a divulgação e incentivo a utilização do sistema de rotas e sucessivamente a sua liberação para utilização da comunidade local e turistas.



Figura 10 – Estrutura da página web
 Fonte: Elaboração Própria.

4 RESULTADOS E DISCUSSÕES

A Base de Dados de Entrada (BDE) foi estabelecida mediante a aplicação do modelo matemático da Fórmula de Haversine nas latitudes e longitudes informados na Tabela 1 e evidenciados no mapa urbano de Iporá, Figura 1. Este conjunto de dados serve como ponto de partida crucial para a aplicação dos algoritmos, fornecendo as informações essenciais para a criação de rotas precisas, ou caminhos mínimos. Este BDE é determinado com base nas distâncias entre cada localidade, conforme destacado na Tabela 3.

Tabela 3 – Base de Dados de Entrada - BDE

local/local	1	2	3	4	5	6	7	8	9	10
1	0	1507	1903	2050	2481	3034	2677	3661	3855	4677
2	1507	0	454	667	1265	1742	1194	2558	2421	3822
3	1903	454	0	229	851	1294	959	2136	1976	3439
4	2050	667	229	0	622	1074	1012	1907	1806	3212
5	2481	1265	851	622	0	557	1389	1292	1476	2592
6	3034	1742	1294	1074	557	0	1491	918	965	2307
7	2677	1194	959	1012	1389	1491	0	2404	1657	3797
8	3661	2558	2136	1907	1292	918	2404	0	1326	1395
9	3855	2421	1976	1806	1476	965	1657	1326	0	2573
10	4677	3822	3439	3212	2592	2307	3797	1395	2573	0

Ao catalogar o desempenho dos resultados provenientes dos algoritmos *Ant Colony Optimization* (ACO), *Bat Algorithm* (BA), *Bee Colony Optimization* (BCO), *Cuckoo Search* (CS), *Firefly Algorithm* (FA), *Genetic Algorithm* (GA) e *Particle Swarm Optimization* (PSO), foi possível analisar não apenas o tempo médio de execução, mas também a eficácia média da geração de rotas obtida durante a execução de cada algoritmo. Esse enfoque proporcionou uma visão abrangente e detalhada sobre a eficiência temporal e a eficácia de geração de rotas de cada implementação.

4.1 Eficiência Temporal

Ao conduzir os testes de execução dos algoritmos para avaliar a eficiência temporal em diferentes máquinas, com suas configurações especificadas em detalhe na mostrados na Tabela 2, registrou-se o tempo médio obtido por cada máquina executando cada algoritmo selecionado. A análise dos resultados revelou padrões distintos de desempenho entre as máquinas Tabela 4, destacando, em alguns casos, diferenças significativas nos tempos de execução em milissegundos dos algoritmos, Figura 11.

Tabela 4 – Tempo médio obtido em cada algoritmo por máquina

Máquina	Algoritmos						
	ACO	BA	BCO	CS	FA	GA	PSO
1	0.03	0.12	0.70	0.05	0.05	1.35	3.71
2	0.06	0.11	0.60	0.03	0.07	1.14	2.98
3	0.05	0.15	0.88	0.04	0.04	1.70	3.72

Tempo médio (ms)

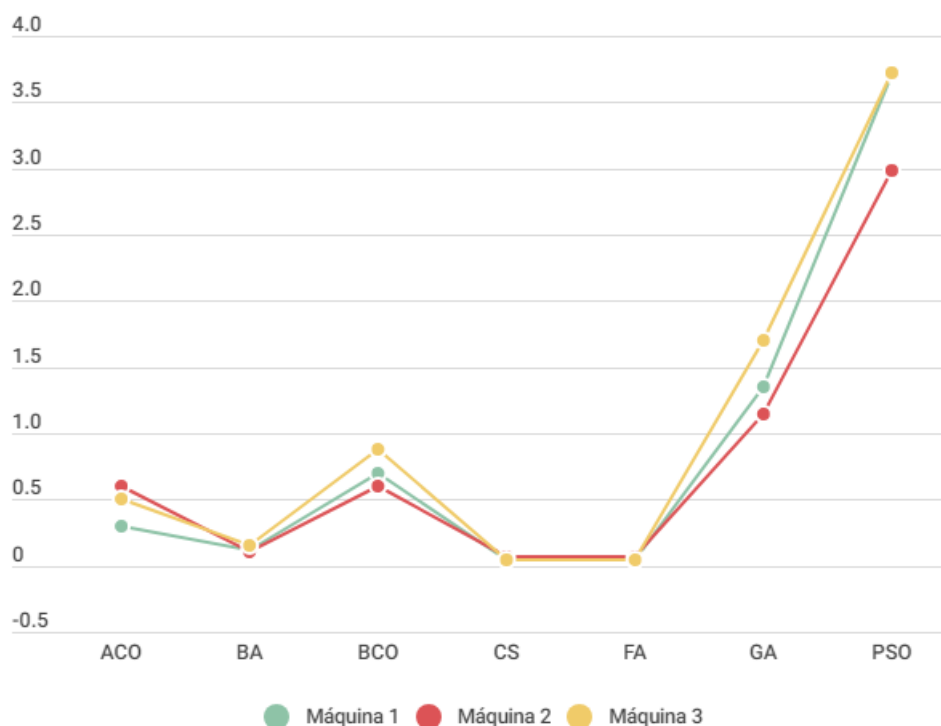


Figura 11 – Gráfico de tempo médio de execução
Fonte: Elaboração Própria.

4.2 Eficácia dos Resultados

Ao conduzir os testes de execução dos algoritmos nas máquinas, Tabela 2, para avaliação da eficácia da construção de rotas através da distância média obtida em metros, Tabela 5. A análise minuciosa dos dados revela variações notáveis nas distâncias médias percorridas, indicando diferentes níveis de eficácia na construção de rotas para cada algoritmo, Figura 12.

Tabela 5 – Distância média obtido em cada algoritmo

Máquina	Algoritmos						
	ACO	BA	BCO	CS	FA	GA	PSO
1	12038	12853	12555	13618	15840	13490	12485
2	12138	12862	12555	13902	15840	13491	12268
3	10652	12851	12555	13459	15840	13492	12082

Distância média (m)

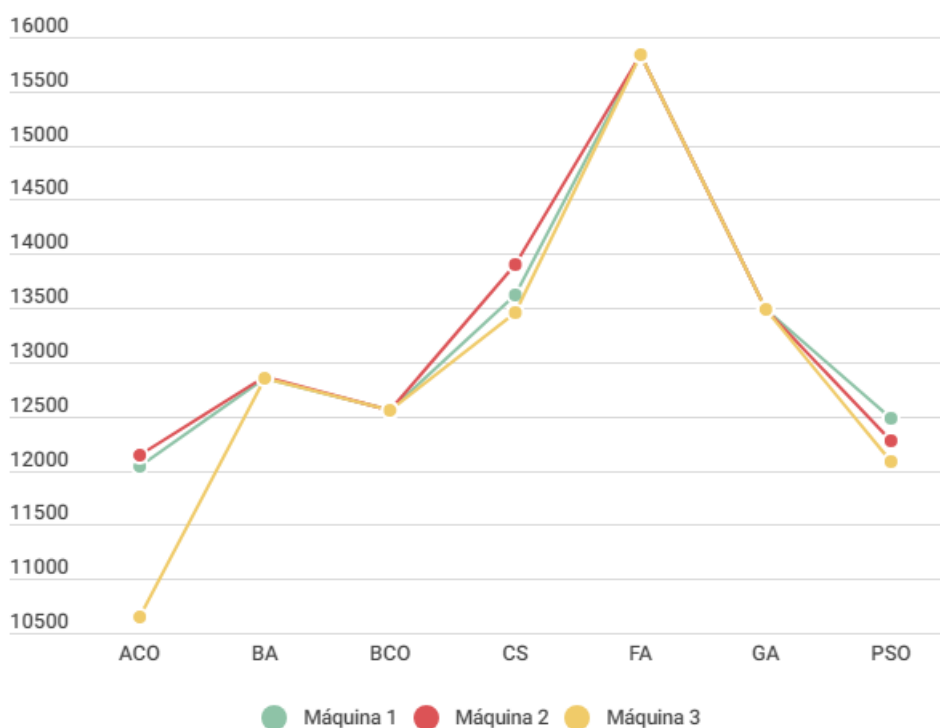


Figura 12 – Gráfico de distância média
Fonte: Elaboração Própria.

4.3 Limitações do Estudo

Com o propósito de identificar e listar uma ampla gama de fatores que podem exercer influência sobre o desempenho da execução de um algoritmo. Portanto, este trabalho possui áreas que requerem considerações adicionais. O reconhecimento dessas limitações é crucial para interpretar os resultados com precisão e para fornecer uma base transparente para pesquisas futuras.

Algumas das limitações evidenciadas durante a execução deste trabalho foram:

- Recursos de Hardware, computadores com hardware mais poderoso provavelmente executarão algoritmos mais rapidamente;

- Eficiência da Implementação, uma implementação em uma linguagem de programação eficiente pode ajudar a executar com mais eficiência;
- Sistema Operacional, a alocação de recursos, escalonamento de processos e outras operações do sistema podem influenciar diretamente o desempenho do algoritmo;
- Ambiente de Execução, outros processos em execução no sistema e consumo excessivo de recursos por parte de outros aplicativos podem influenciar a performance do algoritmo.

4.4 Escolha do Algoritmo

Portanto, considerando os resultados de eficiência e eficácia obtidos da execução dos algoritmos selecionados, o algoritmo *Ant Colony Optimization* obteve menores resultados de tempo de execução e distância média. Essa superioridade nos indicadores de desempenho sugere que o algoritmo *Ant Colony Optimization* demonstrou uma notável capacidade de encontrar soluções mais rápidas e eficientes em comparação com os demais algoritmos testados. Estes resultados positivos destacam a viabilidade e promissora aplicabilidade do algoritmo *Ant Colony Optimization* em contextos nos quais a eficiência operacional, como a minimização do tempo de execução e da distância percorrida, são critérios essenciais.

5 CONSIDERAÇÕES FINAIS

Neste capítulo, integraremos os resultados de eficiência e eficácia dos algoritmos selecionados ao longo desta pesquisa no desenvolvimento de um módulo web destinado à geração de rotas otimizadas para o Museu Virtual de Iporá-GO. Essa abordagem visa traduzir as descobertas e avanços obtidos na análise dos algoritmos em uma aplicação prática, proporcionando uma contribuição tangível para a otimização do percurso no contexto específico do Museu Virtual. Através da conclusão da implementação de um sistema um gerador de rotas otimizadas através do uso de algoritmos bioinspirados e técnicas de inteligência artificial para percorrer todas as localidades históricas informadas no Museu Virtual Total de Iporá — GO.

Mediante o alcance das metas estabelecidas, foi possível concretizar o desenvolvimento do módulo de geração de rotas otimizadas, através do uso do *Ant Colony Optimization* implementado na linguagem *JavaScript*. Este avanço decorre da conclusão de objetivos específicos, incluindo a catalogação de algoritmos bioinspirados, a criação da base de dados para os algoritmos, a realização dos testes de eficiência e eficácia, a comparação dos resultados obtidos, a seleção do melhor algoritmo para o escopo do trabalho e, por fim, a implementação do módulo de geração de rotas, conforme apresentado nas Figuras 13 e 14.

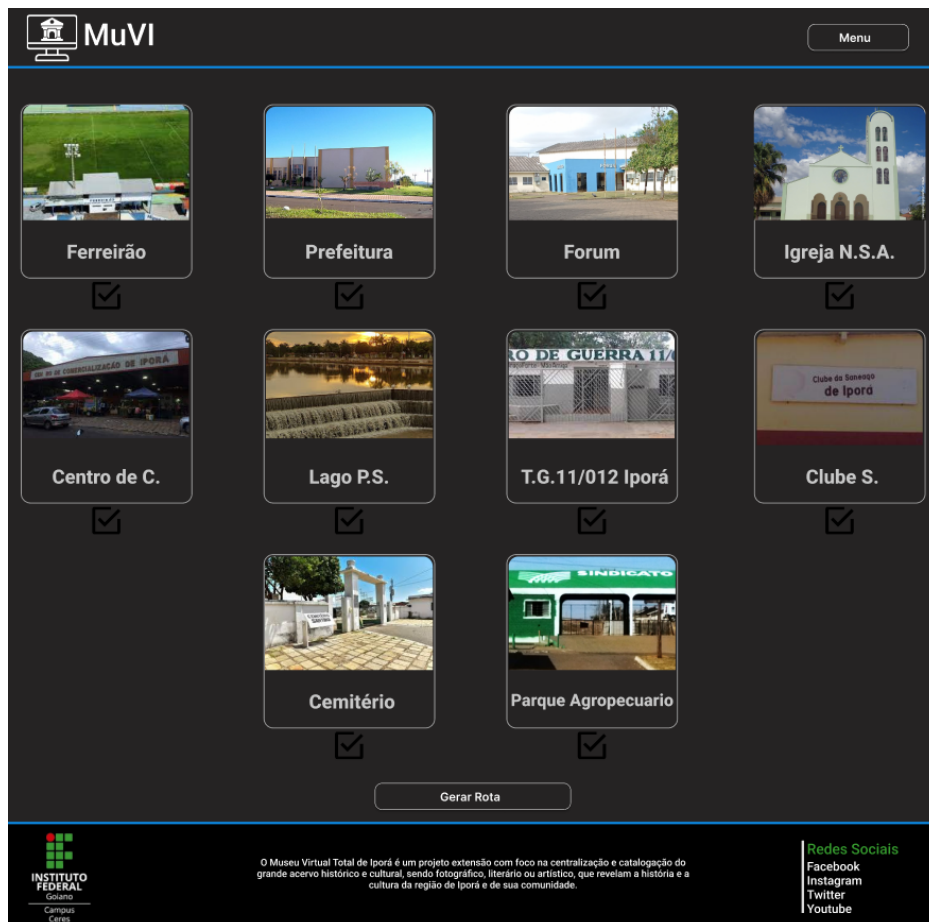


Figura 13 – Módulo de geração de rotas
Fonte: Elaboração Própria.

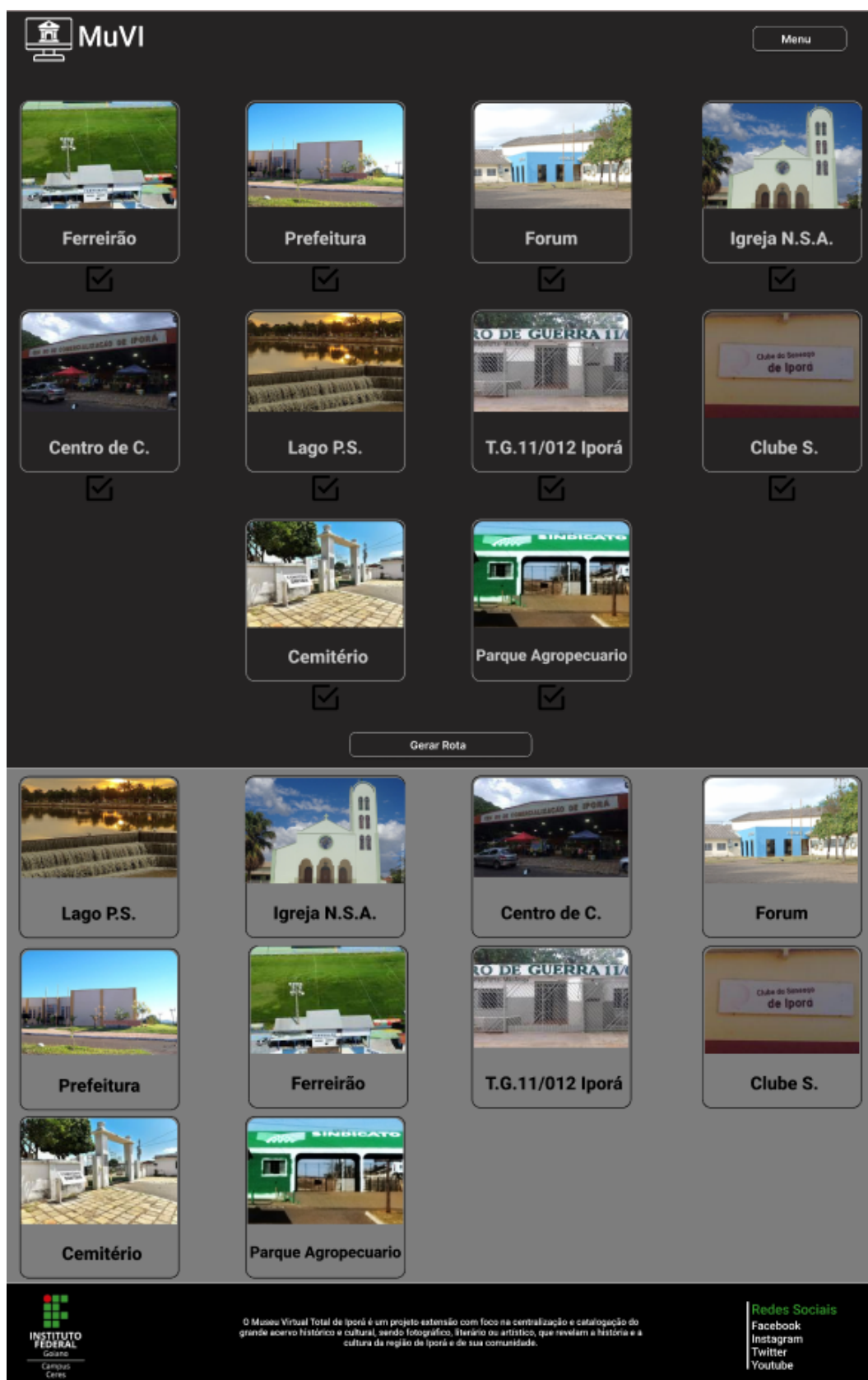


Figura 14 – Geração de rotas
 Fonte: Elaboração Própria.

Devido à ausência de acessos e demanda para o Museu Virtual de Iporá-GO, que permanece desabilitado e inoperante desde 2021, a integração do módulo de geração de rotas ao portal enfrentou desafios. A falta de atividade do museu impacta negativamente nesse processo de incorporação, resultando na necessidade de integração ao Museu Virtual Total de Iporá - GO (MUVI). Em resposta a essa situação, decidiu-se redirecionar o módulo para integrar um projeto de extensão ainda em fase de conclusão no Instituto Federal Goiano — Campus Iporá. Esse projeto tem como propósito a atualização e promoção metodológica do portal, visando, assim, utilizar o módulo como um recurso contributivo para estimular o interesse e a busca pelo Museu Virtual. Essa estratégia visa superar os desafios atuais e alinhar o módulo de geração de rotas a um contexto mais propício para sua efetiva utilização e impacto.

5.1 Trabalhos Futuros

5.1.1 Ampliação do Escopo

Ao considerar o futuro da pesquisa em algoritmos bioinspirados para a geração de rotas otimizadas, nota-se viabilidade da aplicação destes algoritmos com uma Base de Entrada maior. Um estudo crucial pode se concentrar na escalabilidade desses algoritmos, avaliando como seu desempenho se adapta ao aumento do tamanho do banco de dados. Além disso, a otimização específica dos parâmetros dos algoritmos para ambientes com volumes substanciais de dados é uma área promissora, visando garantir eficiência e eficácia.

Avaliar a robustez dos algoritmos em cenários com bancos de dados heterogêneos é fundamental para sua aplicação prática. Da mesma forma, investigar algoritmos bioinspirados específicos, como os baseados em enxame, e algoritmos genéticos, pode fornecer uma compreensão mais profunda de suas características e desempenho em larga escala, sendo uma oportunidade valiosa para aprimorar a aplicação de algoritmos bioinspirados na geração de rotas em ambientes com grandes volumes de dados, contribuindo significativamente para o avanço dessa área específica.

5.1.2 Aprimoramento de Hardware

Futuras pesquisas podem se concentrar em avaliar os algoritmos bioinspirados para geração de rotas em um contexto de hardware mais avançado. A análise de desempenho em hardware de última geração possibilitará uma compreensão mais precisa do potencial desses algoritmos. Uma abordagem promissora seria otimizar os algoritmos para se alinhar às características específicas de arquiteturas mais avançadas, explorando adaptações para aproveitar recursos como arquiteturas paralelas e caches maiores.

Considerar a execução dos algoritmos em ambientes de computação em nuvem oferece a oportunidade de explorar a eficácia em escalas variáveis e sob demanda, aproveitando recursos escaláveis disponíveis. Além disso, a análise do consumo de energia e recursos do hardware é vital, fornecendo uma perspectiva completa da eficiência dos algoritmos em contextos nos quais a eficiência energética é crucial. Proporcionando uma base sólida para aplicações mais eficientes e eficazes em futuros cenários computacionais.

5.1.3 Incorporação de Tecnologias Emergentes

Futuras pesquisas na implementação de algoritmos bioinspirados para a geração de rotas oferecem oportunidades promissoras para aprimorar o desempenho e a relevância dessas soluções, especialmente quando integradas com tecnologias emergentes. Uma direção de pesquisa valiosa envolve a fusão desses algoritmos com técnicas de Inteligência Artificial (IA), como aprendizado de máquina e redes neurais, visando uma adaptação mais eficiente e aprendizado contínuo na geração de rotas.

Outra opção é a adaptação desses algoritmos bioinspirados para otimização em dispositivos móveis e *edge computing* permite uma execução eficiente e responsiva, especialmente em ambientes móveis. Aproveitar as capacidades avançadas de comunicação proporcionadas pela tecnologia 5G também é crucial, possibilitando uma transmissão eficiente de dados entre dispositivos. Oferecendo soluções mais adaptáveis, eficientes e alinhadas com as demandas dinâmicas de ambientes contemporâneos.

Referências

- ARARUNA, A. R. Caminho mínimo com restrição probabilística de atraso máximo. 2013. Citado na página 3.
- ARTZI, S. et al. A framework for automated testing of javascript web applications. In: *Proceedings of the 33rd International Conference on Software Engineering*. [S.l.: s.n.], 2011. p. 571–580. Citado na página 2.
- AZIS, N. et al. How smart tourism technologies affect tourist destination loyalty. *Journal of Hospitality and Tourism Technology*, Emerald Publishing Limited, v. 11, n. 4, p. 603–625, 2020. Citado na página 1.
- BAGGIO, R.; MICERA, R.; CHIAPPA, G. D. Smart tourism destinations: a critical reflection. *Journal of Hospitality and Tourism Technology*, Emerald Publishing Limited, v. 11, n. 3, p. 407–423, 2020. Citado na página 11.
- BARBOSA, C. E. M. *Algoritmos bio-inspirados para solução de problemas de otimização*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2017. Citado na página 2.
- BARREIRA, N. M. C. *Sistema Inteligente para Otimização de Rotas*. Tese (Doutorado), 2016. Citado na página 3.
- BECCENERI, J. C. et al. Otimização por colônia de formigas (ant colony optimization). *Disponível em mtc-m19. sid. inpe. br/col/sid. inpe. br/mtc-m19*, v. 80, n. 2010, p. 01–20, 2010. Citado na página 4.
- BHATT, M. et al. Nature inspired route optimization in vehicular adhoc network. p. 447–451, 2016. Citado na página 11.
- CABUGUEIRA, A. A importância económica do turismo. *Revista Turismo & Desenvolvimento*, Universidade de Aveiro, v. 2, n. 2, p. 97–104, 2005. Citado na página 1.
- CORMEN, T. H. et al. *Introduction to algorithms*. [S.l.]: MIT press, 2022. Citado na página 3.
- DEVILE, E. L. O desenvolvimento do turismo acessível: dos argumentos sociais aos argumentos de mercado. *Revista Turismo & Desenvolvimento*, n. 11, p. 39–46, 2009. Citado na página 1.
- FISTER, I. et al. A comprehensive review of firefly algorithms. *Swarm and evolutionary computation*, Elsevier, v. 13, p. 34–46, 2013. Citado na página 8.
- FOURNIER, P. F. M. *Otimização de rotas: planeamento em viaturas pesadas*. Tese (Doutorado), 2022. Citado na página 3.
- GANDOMI, A. H.; YANG, X.-S.; ALAVI, A. H. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with computers*, Springer, v. 29, p. 17–35, 2013. Citado na página 7.

- GAO, S. *Bio-Inspired Computational Algorithms and Their Applications*. [S.l.]: BoD–Books on Demand, 2012. Citado na página 3.
- GAVALAS, D. et al. Scenic route planning for tourists. *Personal and Ubiquitous Computing*, Springer, v. 21, p. 137–155, 2017. Citado na página 1.
- JEONG, M.; SHIN, H. H. Tourists’ experiences with smart tourism technology at smart destinations and their behavior intentions. *Journal of Travel Research*, SAGE Publications Sage CA: Los Angeles, CA, v. 59, n. 8, p. 1464–1477, 2020. Citado na página 11.
- JÜNGER, M.; REINELT, G.; RINALDI, G. The traveling salesman problem. *Handbooks in operations research and management science*, Elsevier, v. 7, p. 225–330, 1995. Citado na página 11.
- KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: IEEE. *Proceedings of ICNN’95-international conference on neural networks*. [S.l.], 1995. v. 4, p. 1942–1948. Citado na página 9.
- KOCHENDERFER, M. J.; WHEELER, T. A. *Algorithms for optimization*. [S.l.]: Mit Press, 2019. Citado na página 1.
- LUGER, G. F. *Inteligência Artificial-: Estruturas e estratégias para a solução de problemas complexos*. [S.l.]: Bookman, 2004. Citado na página 3.
- MA, X. Intelligent tourism route optimization method based on the improved genetic algorithm. p. 124–127, 2016. Citado na página 11.
- NEVES, M. C. D. et al. Medindo o raio da terra: Uma experiência no ensino. Citado na página 2.
- PACHECO, M. A. C. et al. Algoritmos genéticos: princípios e aplicações. *ICA: Laboratório de Inteligência Computacional Aplicada. Departamento de Engenharia Elétrica. Pontifícia Universidade Católica do Rio de Janeiro. Fonte desconhecida*, v. 28, 1999. Citado na página 9.
- PAINE, L. *The sea and civilization: a maritime history of the world*. [S.l.]: Atlantic Books Ltd, 2014. Citado na página 1.
- SINNOTT, R. W. Virtues of the haversine. *Sky and telescope*, v. 68, n. 2, p. 158, 1984. Citado na página 10.
- SOUFFRIAU, W.; VANSTEENWEGEN, P. Tourist trip planning functionalities: state-of-the-art and future. In: SPRINGER. *10th International conference on Web Engineering (ICWE 2010)*. [S.l.], 2010. v. 6385, p. 474–485. Citado na página 1.
- STASTNÝ, J.; SKORPIL, V.; CIZEK, L. Traveling salesman problem optimization by means of graph-based algorithm. p. 207–210, 2016. Citado na página 1.
- TEODOROVIC, D. et al. Bee colony optimization: principles and applications. In: IEEE. *2006 8th seminar on neural network applications in electrical engineering*. [S.l.], 2006. p. 151–156. Citado na página 6.
- TOWNER, J.; WALL, G. History and tourism. *Annals of Tourism Research*, Elsevier, v. 18, n. 1, p. 71–84, 1991. Citado na página 1.

- XIONG, J. Optimization model of tourist transportation route based on multi-neuron algorithm. p. 1161–1164, 2022. Citado na página 11.
- YANG, X.-S.; HE, X. Bat algorithm: literature review and applications. *International Journal of Bio-inspired computation*, Inderscience Publishers Ltd, v. 5, n. 3, p. 141–149, 2013. Citado na página 5.
- YOO, C. W. et al. Improving travel decision support satisfaction with smart tourism technologies: A framework of tourist elaboration likelihood and self-efficacy. *Technological Forecasting and Social Change*, Elsevier, v. 123, p. 330–341, 2017. Citado na página 1.
- ZHANG, W. et al. Multiobjective particle swarm optimization with improved selection strategy for route optimization. p. 205–209, 2019. Citado na página 11.