

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO
FACULDADE DE CIÊNCIAS EXATAS
DEPARTAMENTO DE COMPUTAÇÃO

PROTÓTIPO DE SISTEMA DE ROTEAMENTO PARA VEÍCULOS
COLETIVOS COMPARTILHADOS

Caroline C. Appolinario
Hector Rauer
João E. L. Fouyer

São Paulo
2019

CAROLINE C. APPOLINARIO , HECTOR RAUER , JOÃO E. L. FOUYER

PROTÓTIPO DE SISTEMA DE ROTEAMENTO PARA VEÍCULOS COLETIVOS COMPARTILHADOS

Trabalho de Conclusão de Curso apresentado à
Pontifícia Universidade Católica de São Paulo,
como um dos requisitos para a conclusão do Curso
de Bacharelado de Ciência da Computação.

Orientador: Prof. Dr. Carlos Eduardo de Barros
Paes

São Paulo
2019

SUMÁRIO

1	INTRODUÇÃO	5
1.1	Motivação	5
1.2	Objetivos	7
1.2.1	Objetivo Geral	7
1.2.2	Objetivos Específicos	7
1.3	Justificativa	8
1.4	Delimitação do Problema	9
1.5	Método de Trabalho	10
1.6	Organização do Texto	11
1.7	Cronograma	12
2	REVISÃO BIBLIOGRÁFICA	13
2.1	Fundamentação Teórica	13
2.1.1	Transporte Coletivo Urbano	13
2.1.2	Problemas de Otimização	16
2.1.2.1	<i>Vehicle Routing Problem (VRP)</i>	16
2.1.3	NP-Completo e NP-Difícil	17
2.1.4	O Problema do Caixeiro Viajante	18
2.1.5	Técnicas Computacionais para Resolução de Problemas de Otimização Combinatória	19
2.1.6	Algoritmos Gulosos	19
2.1.7	Métodos Exatos	20
2.1.8	Métodos heurísticos	20
2.1.9	Métodos meta-heurísticos	20
2.1.9.1	<i>Métodos meta-heurísticos citados pela literatura no VRP</i>	22
2.1.10	Sistemas de Georreferenciamento	23
2.1.11	Agrupamento de Passageiros Candidatos à Rota Através de Grafos de Similaridade e Spectral Clustering	24
2.2	Método de Desenvolvimento	25
2.2.1	Iconix	25
2.3	Trabalhos Relacionados	26
2.3.1	Uma Metodologia para Roteamento de Veículos Escolares Utilizando Sistemas de Informação Geográfica (ROSA, 2018)	26
2.3.2	A survey of models and algorithms for optimizing shared mobility (MOURAD; PUCHINGER; CHU, 2019)	28
2.3.3	Algoritmos de Solução para o Problema do Caixeiro Viajante com Passageiros e Quota (SILVA, 2017)	30
3	DESENVOLVIMENTO	
4	RESULTADOS OBTIDOS	
5	CONCLUSÃO	

REFERÊNCIAS	31
ANEXO	36

1 INTRODUÇÃO

O tempo gasto no trânsito para a locomoção diária é um grande problema para os habitantes que residem em grandes metrópoles. Na região metropolitana de São Paulo, por exemplo, o congestionamento e o tempo gasto no trânsito é a quarto problema mais citado pelos cidadãos (REDE NOSSA SAO PAULO, IBOPE, 2018). Além disto, o transporte público se mostra ineficiente e causa desgaste para seus passageiros frequentes, sendo o quinto problema com o maior número de queixas (REDE NOSSA SAO PAULO, 2015). Com pouca manutenção, muitas vezes os veículos da Grande São Paulo falham em oferecer uma boa experiência de locomoção para seus usuários, principalmente por causa da superlotação nos horários de pico e da falta de conforto dos veículos (REDE NOSSA SAO PAULO, IBOPE, 2018).

Além do problema enfrentado pelos passageiros, o Poder Público também tem o desafio de minimizar os gastos com a Secretaria de Transportes Metropolitanos ao passo que precisa diminuir o trânsito da cidade. Nos últimos anos, novas formas de locomoção foram incorporadas no cotidiano do paulistano a partir de iniciativas privadas ou de parcerias público-privadas (PPP) com ajuda da tecnologia digital, como por exemplo os aplicativos de carona, compartilhamento de bicicletas e patinetes elétricos e o fretamento privado de veículos coletivos – como ônibus e vans.

A evolução da tecnologia da informação, sobretudo da tecnologia móvel, desempenha um papel importante na viabilidade destes serviços, permitindo a interação entre o fornecedor e o consumidor mesmo que não se conheçam. Nos aplicativos de carona, por exemplo, passageiros e motoristas que nunca se viram conseguem ser conectados em tempo real graças à popularização dos *smartphones* e da Internet móvel.

Este trabalho apoia-se na capacidade que a tecnologia da informação tem para potencializar e facilitar o planejamento de rotas para veículos coletivos a partir de dados disponíveis.

1.1 Motivação

O congestionamento na locomoção urbana nas grandes metrópoles tornou-se um problema para seus habitantes e um desafio para gestores públicos. A duração total do deslocamento diário na cidade de São Paulo vem diminuindo nos últimos anos, junto com a utilização de transporte coletivo, reforçando a percepção é de que medidas provisórias são

adotadas como paliativo para problemas do transporte público que nunca são resolvidos.

Mesmo com sensível redução nos tempos de deslocamento dentro da cidade, as demandas por melhorias na lotação e no conforto; nos tempos de espera e de viagem de transporte coletivo permanecem latentes, geram descontentamento e impactam diretamente a avaliação do sistema de transporte público da cidade e na disposição em utilizá-lo.

(REDE NOSSA SAO PAULO, IBOPE, 2018, p. 36)

O transporte coletivo surge com diversos benefícios, como por exemplo uma alternativa que visa diminuir os problemas de congestionamento e os custos com combustível. Entretanto, alguns desafios dificultam ampla adoção de transporte coletivo, como o conforto que o veículo individual oferece ao acesso *door-to-door* (FURUHATA et al., 2013), e a oferta e qualidade dos serviços no transporte público.

Um sistema de roteamento para o transporte coletivos de passageiros oferece a capacidade de minimizar a quantidade de assentos livres em um veículo com o objetivo de diminuir o número de veículos utilizados no transporte de passageiros e consequentemente o congestionamento nas grandes cidades. (MOURAD; PUCHINGER; CHU, 2019)

Paulistanos levam, em média, 1h57 no deslocamento diário para realizar a principal atividade do dia, como por exemplo o trabalho, escola ou faculdade. Considerando apenas passageiros de transporte público, este tempo aumenta para 2h08. Caso todas as atividades do dia sejam consideradas, a média da duração de deslocamento muda para 2h43 no geral, e 2h57 para usuários de transporte público (REDE NOSSA SAO PAULO, IBOPE, 2018). A atual configuração da mobilidade urbana também força passageiros a utilizarem mais de um modal em cada viagem, aumentando o cansaço dos seus usuários frequentes.

Outro desafio no planejamento urbano é atualizar as rotas e a frequência do transporte público conforme a demanda de um destino for alterado. Em média, o tempo de espera por um ônibus municipal na cidade de São Paulo é de 18 minutos (REDE NOSSA SAO PAULO, IBOPE, 2018), sendo a principal preocupação entre as pessoas que não utilizam o transporte, junto à superlotação e a alta tarifa pelo serviço prestado. A falta de planejamento em melhorar os problemas existentes no transporte coletivo, expõe em comparação a pesquisa dos anos anteriores, que não há mudanças significativas no transporte, configurando uma das causas dos altos custos operacionais e mantendo o descaso com seus usuários. (REDE NOSSA SAO PAULO, IBOPE, 2018).

A evolução da tecnologia da informação, sobretudo da tecnologia móvel e da Internet das Coisas (*IoT*), possibilita que ferramentas auxiliares sejam utilizadas para coletar dados e realizar um planejamento mais adequado para transporte coletivo. Desta forma, rotas ociosas poderiam ser alteradas para diminuir a duração total do trajeto e diminuir os custos de operação, assim como novas rotas poderiam ser criadas para atender demandas existentes e melhorar a abrangência do serviço ofertado. Com ajuda das soluções tecnológicas já existentes como *Google Maps* e *Waze*, é possível ampliar as funcionalidades que um simples *GPS* pode proporcionar, fazendo com que a rota seja calculada para atender não só uma pessoa individual, mas várias delas ao mesmo tempo, levando em consideração o deslocamento e tempo de viagem.

A tecnologia da informação e métodos computacionais podem servir de apoio para ferramentas que visam facilitar e ampliar o planejamento das rotas de transporte coletivo de pessoas em uma metrópole como São Paulo, diminuindo a dificuldade de gestores de transportes públicos e de administradores de empresas de fretado em planejar essas rotas para os veículos que as realizarão. Neste trabalho, será desenvolvido um protótipo que aplica e adapta conjuntos de métodos computacionais para aprimorar o roteamento de veículos para o transporte de passageiros, seguindo critérios e restrições que serão detalhadas nos objetivos e delimitação do problema.

1.2 Objetivos

Nas subseções a seguir, serão apresentados o objetivo geral e os objetivos específicos deste trabalho.

1.2.1 Objetivo Geral

Desenvolver um protótipo de roteamento de veículos para o transporte de passageiros utilizando métodos de solução de problemas que já foram empregados neste contexto. Levando em consideração que o Problema do Caixeiro Viajante abrange estas classes de soluções, ele será utilizado como ponto de partida para a formulação do protótipo.

1.2.2 Objetivos Específicos

Nos objetivos específicos deste trabalho pretende-se:

- Agrupar passageiros que possam convergir em uma rota a fim de minimizar o número

de assentos livres por viagem e respeitando a capacidade do veículo;

- Indicar os pontos de embarque e desembarque de cada passageiro respeitando o limite máximo de deslocamento necessário a partir da origem/destino até o ponto de embarque e a partir do ponto de desembarque até a origem/destino;
- Ordenar os pontos de embarque e desembarque a fim de diminuir o tempo do trajeto e a distância percorrida pelo veículo;
- Formular um protótipo para buscar a distância e o tempo de deslocamento real entre os pontos de embarque e desembarque;
- Definir os horários de embarque e desembarque dos pontos de acordo com a estimativa de tempo, respeitando a janela de horário de cada passageiro;
- Atualizar e adequar a rota de acordo com entrada e a saída de passageiros.
- Validar o modelo desenvolvido por meio de método computacional, em um problema real.

1.3 Justificativa

A partir da Emenda Constitucional número 90 de 2015 (BRASIL, 2015), o transporte é considerado um direito social de todo cidadão brasileiro, junto com educação, saúde e previdência social. Apesar disto, o sistema de transporte público ainda é deficiente em diversos municípios brasileiros, mesmo apresentando despesas altas aos cofres públicos e nas tarifas pagas pelos seus usuários.

No caso do município de São Paulo, o valor orçado para a secretaria de transportes urbanos em 2019 foi de aproximadamente R\$ 3,4 bilhões (LOA, 2018). O alto investimento não é justificado em melhoria na qualidade do serviço, mas pode ser atribuído a diversos fatores, como a falta de trabalhadores qualificados e de recursos necessários para planejar e gerenciar a rede de transporte, que inclui estabelecer o itinerário e horário das rotas de ônibus de acordo com a demanda, designar e atribuir os veículos às rotas, considerando o custo, a segurança, o tempo de deslocamento e a capacidade dos veículos (LIMA et al., 2016).

Por outro lado, ambos Governo do Estado e da Cidade de São Paulo, facilitam a coleta e contextualização de informações referentes ao transporte e locomoção na região

metropolitana de São Paulo através de pesquisas realizadas e de ferramentas disponíveis para demais pesquisadores. São exemplos a pesquisa de Origem e Destino realizada pelo Metrô de São Paulo (METRÔ, 2012), a interface para acesso à informação para desenvolvedores fornecida pela SPTrans (SPTRANS, 2019) e a composição do estudo tarifário que é realizado anualmente (SPTRANS, 2018). Além disto, em face dos desafios de mobilidade urbana e do esforço coletivo em superá-los, o Governo do Estado criou um programa exclusivo de Parceria Público-Privada (PPP) para mobilidade na região metropolitana (GESP, 2015).

Este esforço coletivo também é evidente em organizações não-governamentais e em setores privados. A Rede Nossa São Paulo, por exemplo, possui pesquisas que tangem o assunto sob diferentes aspectos (RNSP, 2018). São Paulo é a única cidade brasileira no Movement (UBER, 2019), plataforma da Uber que disponibiliza para gestores públicos e pesquisadores informações detalhadas sobre o tempo de locomoção de acordo com as viagens realizadas no serviço.

Portanto, o problema que emerge deste trabalho é de como a tecnologia da informação pode servir de ferramenta para auxiliar no planejamento de rotas para veículos coletivos. O instrumento resultante poderia ser utilizado por gestores públicos e/ou pela iniciativa privada ao desenvolver formas alternativas de transporte coletivo. A escolha de São Paulo se deve, principalmente, à disponibilidade de dados e de pesquisas que podem ser utilizadas no desenvolvimento do trabalho.

1.4 Delimitação do Problema

O estudo do modelo de solução do problema considera apenas a região da Grande São Paulo, devido à disponibilidade de dados de pesquisas anteriores e de ferramentas do setor público e privado. São ignoradas tanto restrições físicas – como a inviabilidade de um veículo passar por uma via estreita – quanto legais – como a Zona de Máxima Restrição de Fretamento (ZMRF) ou licenciamento necessário para um veículo realizar o transporte coletivo de pessoas. O processamento do georreferenciamento do mapa, pontos e das vias será realizado com suporte das bibliotecas *Places API*, *Directions API*, *Distance Matrix API*, *Geocoding API* e *Maps Embed API*, o uso é detalhado na fundamentação teórica.

Visando o conforto em realizar uma viagem inteira sentado, a quantidade de passa-

geiros que podem ser transportados em um veículo será considerada a partir da quantidade de assentos disponíveis. Será levado em consideração que uma viagem deverá ser realizada a partir da origem especificada pelo usuário – normalmente a residência – até o destino – colégio, faculdade ou trabalho – desconsiderando possíveis integrações com outros modais de transporte – ônibus municipais, trens ou metrô – ou outros fretados. Caso algum usuário ache conveniente realizar alguma integração, ele deverá explicitamente definir o endereço de onde a integração é realizada como origem e/ou de destino.

1.5 Método de Trabalho

A pesquisa irá se basear em procedimentos que consistem em estudos bibliográficos que envolve a leitura de livros, artigos acadêmicos, notícias e informações do poder público e organizações não-governamentais para planejamento de mobilidade urbana. A pesquisa também se baseará em estudos de casos, visto que abordará empresas transportes públicos e privados e modelos adotados em outras cidades. Por fim, será realizada a modelagem e a implementação do protótipo e análise dos resultados.

1. **Estruturação do Problema** – Contextualizar e entender os desafios da mobilidade urbana e as soluções que já são utilizadas pela engenharia de tráfego para contorná-los. Grandes metrópoles como São Paulo contam com inteligência de tráfego para entender melhor problemas de trânsito, demandas e as rotas de transportes coletivos e soluções mais robustas para cada um dos problemas citados. Diversos trabalhos acadêmicos publicados também propõem soluções diversas para melhorar o transporte nas grandes cidades em contextos diferentes.
2. **Revisão Bibliográfica** – Estudar métodos para soluções de problemas de mobilidade urbana, logística, heurística e meta-heurística que já foram publicados e contextualizá-los com o problema deste trabalho. Entender como a sociedade urbana se organiza em rede e como a topologia pode ser aplicada na locomoção diária de uma metrópole como São Paulo. A ideia de rede pode ser analisada a partir da metodologia de agregar insumos de uma mesma natureza ou de naturezas diferentes. “A rede é o próprio tecido constitutivo da sociedade, engendrando desde as redes sociais pessoais” (INOJOSA, 1999). Analisar as diferentes formas de armazenar e indexar pontos e rotas georreferenciadas.

3. **Definição e implementação do protótipo** – Modelar os casos de uso, as variáveis de decisão, as restrições e modelo de relação entre veículos, rotas, motoristas, passageiros e pontos. Definição de tecnologias que serão utilizadas e do método de desenvolvimento do protótipo.
4. **Planejamento e execução de testes** – Criar a massa de testes para medir a eficiência do protótipo com simulações e interpretar seus resultados.
5. **Análise de resultados** – Analisar os resultados fornecidos pelos testes e verificar se cumprem os requisitos do protótipo.

1.6 Organização do Texto

O projeto é apresentado em 5 capítulos, referências bibliográficas e os anexos.

No capítulo 1, Introdução, são apresentados a problemática, motivação, objetivos gerais e específicos e a metodologia de trabalho que será utilizada neste projeto.

No capítulo 2, Revisão Bibliográfica, serão apresentados a Fundamentação Teórica que abordará a complexidade dos problemas de roteamento de veículos, seja no contexto de logística para entregas ou para transporte de passageiros e os trabalhos relacionados ao tema proposto.

No capítulo 3, Desenvolvimento, serão apresentados o trabalho de pesquisa e o processo de desenvolvimento do protótipo.

No capítulo 4, Resultados Obtidos, serão apresentados os testes para aplicação no protótipo e seus resultados, a fim de interpretá-los

Por fim, no capítulo 5, Conclusão, será exposta a análise crítica do problema, os objetivos atingidos e quais são os pontos de melhorias para trabalhos futuros.

1.7 Cronograma

O cronograma é apresentado com a distribuição das atividades propostas no Método de Trabalho 1.5.

Tabela 1 – Cronograma das Atividades

Atividades	fev	mar	abr	mai	jun	jul	ago	set	out	nov
Estruturação do Problema	x	x	x							
Revisão Bibliográfica		x	x	x	x	x				
Definição e Implementação do Protótipo						x	x	x	x	x
Criação e Execução de testes						x	x	x	x	x
Análise de resultados								x	x	x
					Qualificação do TCC					Defesa do TCC

Fonte: Elaborado pelos autores.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo é apresentada a revisão da bibliografia que dará suporte e embasamento para o desenvolvimento deste trabalho, expondo técnicas, conceitos fundamentais e tecnologias que são citadas na literatura para a resolução do problema proposto. Os principais fundamentos teóricos que são cruciais para a compreensão do trabalho são abordados na seção 2.1. A proposta para o método de desenvolvimento do protótipo é discutida na seção 2.2. Por fim, alguns trabalhos relacionados são explorados na seção 2.3, onde são destacadas as semelhanças, diferenças e contribuições para o trabalho.

2.1 Fundamentação Teórica

Nesta seção são apresentados conceitos fundamentais para a compreensão do presente trabalho. Na seção 2.1.1, é apresentada uma contextualização sobre o transporte de passageiros em veículos coletivos urbanos. Na seção 2.1.2 são apresentados problemas de otimização, incluindo um problema clássico de otimização denominado *Vehicle Routing Problem* e suas classes, que são detalhadas na subseção 2.1.2.1.

As principais técnicas computacionais para a resolução de problemas de Otimização Combinatória, incluindo o Problema de Roteamento de Veículos são introduzidas na subseção 2.1.5 classificados em 2.1.3 e exploradas em 2.1.6 (Algoritmos Gulosos) 2.1.7 (Métodos Exatos), 2.1.8 (Métodos Heurísticos) e 2.1.9 (Métodos Meta-Heurísticos). Por fim, sistemas de georreferenciamento citados na literatura são apresentados na subseção 2.1.10

2.1.1 Transporte Coletivo Urbano

Existem diversos atributos na relação do passageiro com o transporte público coletivo, mas esses atributos são independentes e devem ser estudados de forma isolada. A resolução do Contran (DENATRAN, 2015) é feita para que o veículo fique de acordo com cada atributo, sendo eles:

- confiabilidade;
- tempo;
- acessibilidade;

- conforto;
- segurança;
- custo (tarifas).

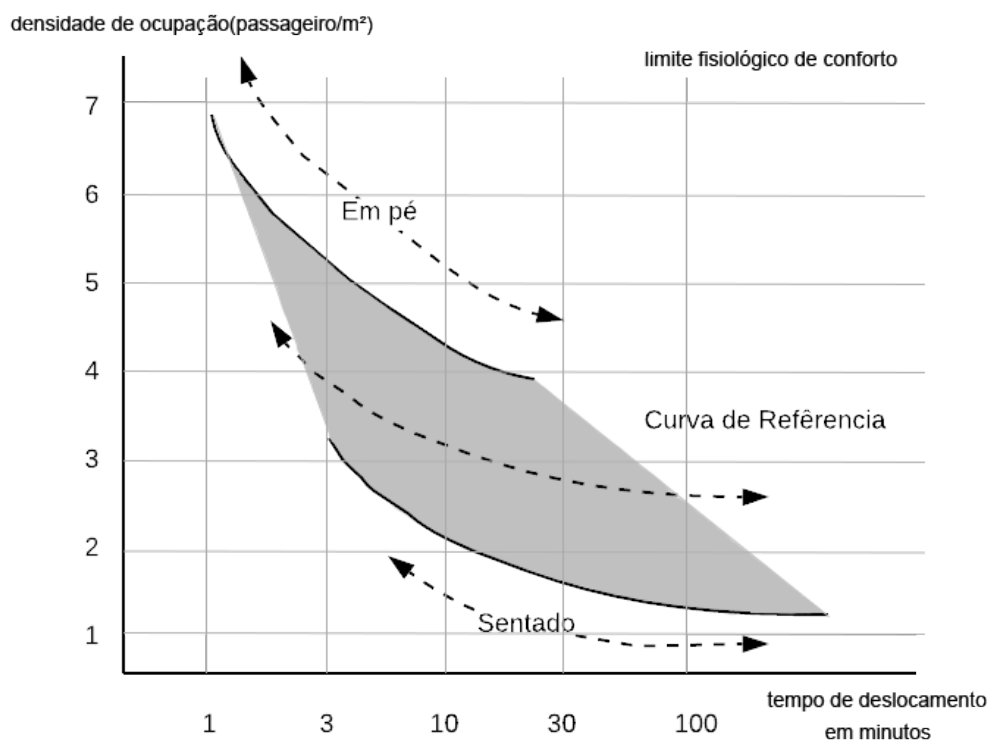
No Anexo A – Tabela de Indicadores de Qualidade do Serviço, consta a tabela 3 que relaciona o atributo do serviço com a sua definição e um indicador equivalente para a sua avaliação.

Apesar da forte relação com o passageiro, as composições homologadas são de viés automotivo e não pensados no lado do usuário.

Os atributos de tempo e capacidade dos veículos caracterizam o transporte urbano coletivo e estão ligados no projeto do veículo, que são prefixados pelos fabricantes. Para o passageiro, o aspecto conforto é composto por um veículo com pouca ocupação, garantindo que o passageiro estará sentado durante a viagem, mas o que também faz parte da satisfação do usuário é a extensão do deslocamento.

A Figura 1 a seguir representa a relação entre a densidade de ocupação de um veículo (em pessoas por m^2) e o tempo de deslocamento (em minutos).

Figura 1 – Relação entre densidade de ocupação e tempo de deslocamento



Fonte: (MARTINS, 2015, p.307)

Assim, em relação ao transporte de passageiros, os princípios que devem estar presentes para um bom nível de serviço, em destaque, são:

- **Permanência:** um serviço constante, sem problemas de prestação de serviços.
- **Eficiência:** oferta de um serviço satisfatório (conforto).
- **Modicidade:** serviços a preços razoáveis

Os atributos mencionados serão utilizados como restrições do projeto, para atingir os objetivos estabelecidos.

- A permanência garante que o serviço prestado será do começo ao fim, sem integração de modais e com horários pré-estabelecidos com o usuário;
- A eficiência visa que o usuário terá seu assento reservado e uma previsão de tempo de chegada definida;

- A modicidade trará o transporte coletivo como o meio de dividir o valor dos custos, como manutenção, entre todos os passageiros e assim, o preço será reduzido em relação ao valor do uso de veículo particular.

2.1.2 Problemas de Otimização

Problemas de otimização podem ser fundamentalmente divididos entre aqueles com variáveis contínuas ou de natureza discreta (PAPADIMITRIOU; STEIGLITZ, 1998). Nos problemas de otimização de variáveis contínuas procura-se por uma função ou um conjunto de números $R \in \mathbb{R}$ para a solução.

Já os problemas de otimização de natureza discreta podem ser solucionados por meio de técnicas de Otimização Combinatória. Nos Problemas de Otimização Combinatória procura-se um conjunto finito de soluções – naturalmente inteiras – e que podem ser formulados como Problemas de Programação Inteira e consequentemente solucionados por meio de Algoritmos de Programação Inteira (GOLDBARG; GOLDBARG; LUNA, 2015) ou podem ser formulados por meio de análise combinatória e solucionados como problemas em Grafos ou algoritmos especificamente desenvolvidos.

Problemas de Otimização Combinatória podem ser formulados como (F, c) onde F é um conjunto de pontos possíveis; e c uma função de custo (DIAS, 2017).

$$c : F \longrightarrow \mathbb{R}^1$$

Onde o problema é encontrar um $f \in F$ para qual

$$c(f) \leq c(y) : y \in F$$

O ponto f é denominado de ótimo global ou solução ótima.

O Problema de Roteamento de Veículos (*Vehicle Routing Problem* - VRP) é um problema clássico de otimização combinatória e que será detalhado na subseção a seguir.

2.1.2.1 *Vehicle Routing Problem (VRP)*

O Problema de Roteamento de Veículos (VRP) foi introduzido na literatura em 1959 por Dantzig e Ramser (TOTH; VIGO, 2002). O VRP tem sido, particularmente, um dos problemas mais abordados nas áreas de Otimização Combinatória e Pesquisa Operacional nas últimas décadas (STEINER et al., 2000). E desde então, vários cenários foram

derivados, como o roteamento para veículos de entrega, roteamento de veículos escolares ou o roteamento de veículos para transporte de passageiros no contexto urbano, que é o objeto de estudo do trabalho presente. De modo geral, consiste em definir rotas para cumprimento de demandas, considerando o número de veículos disponíveis e respeitando a capacidade de cada veículo.

O Problema de Roteamento de Veículos Abertos (*Open Vehicle Routing Problem* – *OVRP*) é uma das existentes derivações do *VRP*. O *OVRP* consiste em caminhos hamiltonianos, ou seja, apresenta um ciclo que percorre todos os vértices do grafo, para um conjunto de clientes atribuídos num mesmo veículo a partir de uma demanda total de todos os clientes de uma rota e que não deve exceder capacidade do veículo. Cada cliente é visitado apenas uma vez por um dos veículos. Porém, o *OVRP* é dado como problema de coleta e entrega de carga e não propriamente de passageiros (BRANDAO, 2004).

No problema em questão, a partir de passageiros coincidentes de uma região, o objetivo é transportá-los para um local próximo ao seu destino, minimizando custos e melhorando frotas para atendimento de regiões condizentes.

Tendo em vista ser um problema de otimização e combinatória, há diversas maneiras de solução, seja com algoritmos gulosos ou por soluções exatas, heurísticas e metaheurísticas. Antes mesmo de entender os tipos de soluções, existe as classes que os algoritmos podem pertencer e, assim, é possível decidir qual deles é mais adequada de acordo com a solução para o seu problema.

Por fim, sabe-se que no contexto do problema de roteamento de veículos, muitas vezes ainda não existem técnicas computacionais capazes de garantir que o caminho encontrado será o menor ou o mais rápido em um tempo polinomial. Desta forma, a solução encontrada pode não ser ótima global, mas sim a local.

2.1.3 NP-Completo e NP-Difícil

A classe NP (*nondeterministic polynomial time*) é a classe de linguagens decidíveis que podem ser verificadas por um algoritmo em tempo polinomial. O problema do Caixeiro Viajante, por exemplo, tem complexidade $\mathcal{O}(n!)$, porém essa complexidade pode ser diminuída até $\mathcal{O}(n^2 2^n)$ (ZIVIANI, 2010, p. 405) com programação dinâmica que é um método para construir algoritmos no contexto de resolver problemas computacionais. Vejamos qual a diferença entre a classe NP-Completo e NP-Difícil. A definição de acordo com

Ziviani é de que “Apenas problemas de decisão (“sim/não”) podem ser *NP-Completo*.” (ZIVIANI, 2010, p. 415)

Assim, se um problema da classe *NP-Completo* pode ser resolvido em tempo polinomial a partir de uma máquina determinística, então todos os problemas da classe podem. Isto é, a classe P , que é constituída pelos problemas de decisão limitados polinomialmente, é igual a classe NP . Essa afirmação ainda não foi comprovada com nenhum problema da classe *NP-Completo*.

Um problema A pertence a classe *NP-Difícil* se existe um problema $B \in NP-Completo$ que pode ser transformado em A em tempo polinomial. Portanto, a única diferença com relação à classe *NP-Completo* é que A não é provado ser NP .

(BUENO, 2011, p. 4)

Concluindo que o problema que pertence à classe *NP-Difícil* não é provado ser resolvido em tempo polinomial. Problemas de otimização e combinatória para o *VRP*, como o problema caixeiro viajante, geralmente, pertencem a classe *NP-Difícil*.

2.1.4 O Problema do Caixeiro Viajante

O problema do Caixeiro Viajante (*Travelling Salesman/Salesperson Problem – TSP*) é um problema clássico de otimização e combinatória e está entre os antigos problemas já estudados na área. Foi formulado matematicamente pela primeira vez em 1932, por Karl Menger (OLIVEIRA, 2015) e possui muitas variações. Este problema pode ser representado por grafo. Seja $G = (V, E)$ um grafo, onde V é um conjunto finito cujos elementos são denominados vértices e E é um conjunto de subconjuntos de dois elementos de V denominados arestas (NOGUEIRA, 2015). Para o TSP as cidades são os vértices e as ligações entre as cidades são as arestas do grafo. Para formulação do problema, dado um conjunto de cidades é “como o caixeiro passaria por todas as cidades, uma única vez com o menor trajeto possível?” A resposta dessa questão, com o caixeiro retornando para a cidade de origem, gera uma ordenação das cidades que o caixeiro deve percorrer. Sua variante mais conhecida é a descrita acima, também conhecido como *TSP com retorno*. Caso não faça parte da solução retornar para a cidade inicial, a variante é conhecida como *TSP sem retorno*, sendo a variante que será utilizada neste trabalho.

2.1.5 Técnicas Computacionais para Resolução de Problemas de Otimização Combinatória

O desenvolvimento da capacidade de processamento dos computadores nos últimos anos tem servido de suporte para a solução – mesmo que parcial – de problemas de Otimização Combinatória, algumas vezes envolvendo centena de milhares de variáveis e restrições.

Diferentes métodos podem ser utilizados para a solução do VRP, que fundamentalmente a literatura os separa entre três principais classes: Métodos Exatos, Métodos Heurísticos e Meta-Heurísticos (ROSA, 2018). Antes mesmo de abordar os tipos de soluções, é necessário entender as classes que os algoritmos podem pertencer e, assim, será possível escolher os melhores candidatos para a solução do problema proposto. Nas subseções seguintes, um levantamento dos métodos mais frequentes para a solução foi realizado a fim de guiar na decisão da escolha da abordagem para resolver o problema proposto pelo trabalho.

2.1.6 Algoritmos Gulosos

Algoritmos com estratégia gulosa (=gananciosa = *greedy*) (FEOFILOFF, 2015) são os relacionados com problemas de otimização que não repensam a solução tomada, verificando a informação disponível no momento sem se preocupar com efeitos futuros. Quando um algoritmo guloso se baseia no princípio da otimalidade e funciona, a primeira solução encontrada será sempre a solução ótima, um exemplo seria um algoritmo para achar o caminho mais curto entre dois pontos. Se o algoritmo não seguir princípio da otimalidade, temos o que é conhecido como heurística, que não é garantindo que a solução encontrada será ótima (ZIVIANI, 2010, p. 59).

Não existe necessidade de avaliar alternativas, nem de empregar algoritmos sofisticados que permitam desfazer decisões tomadas previamente. A razão de o algoritmo ser chamado guloso é que o algoritmo escolhe, a cada passo, o candidato mais evidente que possa ser adicionado à solução.

(ZIVIANI, 2010, p. 58)

2.1.7 Métodos Exatos

Em métodos exatos uma varredura é realizada sobre todo o domínio de busca das soluções, garantindo que o ótimo global seja encontrado sempre que a solução for viável. Entretanto, algoritmos de métodos exatos não podem ser executados em tempo polinomial, de forma que a equação de tempo em função dos número de insumos cresce de forma polinomial (ROSA, 2018).

São apresentados alguns métodos exatos mais citados pela literatura para resolver o VRP no Anexo B – Métodos Exatos Utilizados no VRP.

2.1.8 Métodos heurísticos

A etimologia da palavra *heurística* tem origem do verbo grego *heuriskein* – cuja conjugação na primeira pessoa é *eureka* – que significa descobrir. No entanto, quando a palavra é empregada no contexto de métodos para a solução de problemas de Otimização Combinatória, o seu sentido pode ser ampliado conforme, por exemplo, a definição de Kendra van Wagner-Cherry “*Uma heurística é um atalho mental que permite às pessoas resolver problemas e fazer julgamentos rápida e eficientemente.*” (GOLDBARG; GOLDBARG; LUNA, 2015)

Métodos heurísticos são aqueles que garantem um bom tempo computacional, porém, não garantem uma solução ótima em todos os casos que atuam. No caso de uma aplicação, uma aproximação heurística traz o benefício do tempo de processamento, uma vez que uma solução ótima, com alto tempo de processamento, deixa de ser uma boa solução para o usuário.

São apresentados alguns métodos heurísticos mais citados pela literatura para resolver o VRP no Anexo C – Métodos Heurísticos Utilizados no VRP.

2.1.9 Métodos meta-heurísticos

O prefixo *meta* é de origem grega e seu significado original é *além*. Entretanto, em diversas áreas de conhecimento, o seu uso normalmente é empregado para fornecer um significado sobre algo de sua própria natureza, como um conceito que é uma abstração que explica um outro conceito. Desta forma, *meta-heurísticas* são heurísticas projetadas para encontrar, gerar, ou selecionar heurísticas.

Formalmente, uma meta-heurística é composta por um conjunto de regras estru-

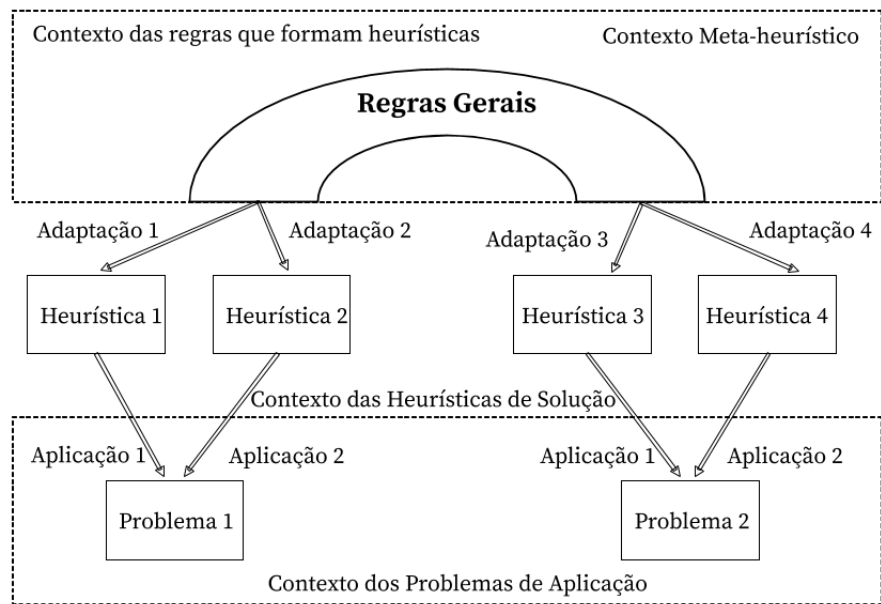
turadas que podem servir de base para a projeção da solução a partir de uma gama de heurísticas computacionais. Estratégias genéricas derivadas de heurísticas podem ser aplicadas em variados contextos de problemas de otimização a partir de adaptações para resolver um problema específico (ROSA, 2018). Muitas meta-heurísticas são inspiradas em fenômenos naturais, como a Colônia de Formigas ou Enxame de Abelhas. Assim como na heurística, a solução ótima não é garantida nos métodos meta-heurísticos. A definição de Goldberg *et al.* (2015) indica o agrupamento de heurísticas computacionais para compor um sistema geral de regras:

Trata-se de uma arquitetura geral de regras que, formada a partir de um tema em comum, pode servir de base para o projeto de uma ampla gama de heurísticas computacionais.

(GOLDBARG; GOLDBARG; LUNA, 2015, p. 75)

A Figura 2 exibe características genéricas de uma abordagem meta-heurística (indicado na figura como Regras Gerais) para agrupar e adaptar heurísticas de determinado tema em comum para a resolução de mais de um problema dentro do contexto dos Problemas de Aplicação.

Figura 2 – Arquitetura geral para meta-heurísticas



Fonte: (GOLDBARG; GOLDBARG; LUNA, 2015, p. 75)

2.1.9.1 Métodos meta-heurísticos citados pela literatura no VRP

Na Tabela 2 são listadas algumas referências de trabalhos acadêmicos relacionados que já utilizaram métodos meta-heurísticos para encontrar a solução.

Tabela 2 – Utilização de meta-heurística para a resolução do VRP

Referência	Tema
(TAILLARD, 1991)	Robust taboo search for the quadratic assignment problem
(TSUBAKITANI; EVANS, 1998)	An empirical study of a new metaheuristic for the traveling salesman problem
(FU; EGLESE; LI, 2005)	Open vehicle routing problem
(MARINAKIS; MARINAKI; DOUNIAS, 2010)	Honey Bees Mating Optimization algorithm for large scale vehicle routing problems
(MARINAKIS; MARINAKI, 2011)	Honey bees mating optimization algorithm for the Euclidean traveling salesman problem
(DIAS et al., 2014)	An Inverted Ant Colony Optimization approach to traffic
(ESCARIO; JIMENEZ; GIRON-SIERRA, 2015)	Ant Colony Extended: Experiments on the Travelling Salesman Problem

Fonte: Elaborado pelos autores.

São detalhados alguns destes métodos meta-heurísticos mais citados pela literatura para resolver o VRP no Anexo D – Métodos Meta-Heurísticos Utilizados no VRP.

2.1.10 Sistemas de Georreferenciamento

Georreferenciamento é a prática de pegar determinado local (ou ponto) no mapa e inserir suas coordenadas em um sistema de referências planas. No contexto de Georreferenciamento, os pontos escolhidos para a extração das coordenadas são chamados de Pontos de Controle. As coordenadas reais dos pontos de controle (verdade terrestre) podem ser obtidas em trabalho de Campo (Lev. Topográfico, GPS, mapas em papel ou digitais georreferenciados) (PANCHER; FREITAS, 2018).

O *Google* fornece uma vasta gama de ferramentas de geolocalização que contribuem para o desenvolvimento de vários aplicativos e softwares que tem como objetivo guiar o usuário de alguma forma à um destino. Dentre elas, temos algumas *APIs* que se destacam quando se tem como objetivo, criar uma plataforma de roteamento veicular, sendo elas:

- **Google Places API** – É um serviço que retorna informações sobre um lugar a partir de requisições HTTP, os lugares são definidos nessa API como estabelecimentos, localizações geográficas ou pontos de interesse proeminentes;
- **Google Directions API** – Fornece rotas de transporte público, bicicleta, carro e a pé. Calcula os tempos de deslocamento atuais ou futuros com base no trânsito em tempo real;
- **Google Distance Matrix API** – Fornece tempos de deslocamento e distâncias para um ou mais locais.;
- **Google Geocoding API** – Transforma dados de endereço de ruas em dados de altitude e latitude;
- **Google Maps Embed API** – Realiza a inclusão de uma interface gráfica de mapeamento da própria Google.

Esses serviços terão um papel importante no desenvolvimento da plataforma de roteamento de transportes coletivos. O uso dessas ferramentas facilita a entrada de informações na plataforma, e gera mais consistência na montagem da parte programática do projeto em questão.

2.1.11 Agrupamento de Passageiros Candidatos à Rota Através de Grafos de Similaridade e Spectral Clustering

Agrupamento (*Clustering*) é uma técnica muito usada hoje em dia, em qualquer área que lida com dados empíricos (valores numéricos concretos). Trabalhadores e pesquisadores que têm essa quantidade de dados em mão, tentam agrupar os dados que tem certo grau de similaridade com o objetivo de manter suas informações organizadas e facilitar o manuseio desses dados. Uma variação desta técnica cujo a performance toma destaque sobre os algoritmos de agrupamento padrão é o *Spectral Clustering*. O *Spectral Clustering* agrupa nós de uma matriz de adjacências (ou grafo de similaridades) e dados considerados similares através do uso de métodos de álgebra linear e cálculo de grafos Laplacianos (LUXBURG, 2007).

O algoritmo de *Spectral Clustering* é dividido nos seguintes passos (SHI; MALIK, 2000):

1. Entrada: Uma matriz de similaridade $S \in \mathbb{R}^{n \times n}$, e o número k de grupos a serem construídos;
2. Construção do grafo de similaridade a partir de pontos X_1, \dots, X_n com noções de similaridade $S_{ij} \geq 0$ entre todos os pontos X_i e X_j .
3. Computar o Laplaciano L não-normalizado.
4. Computar os primeiros k eigenvetores U_1, \dots, U_k da generalização do eigenproblema $Lu = \lambda Du$.
5. Seja $U \in \mathbb{R}^{n \times k}$ a matriz contendo os vetores U_1, \dots, U_k sendo as colunas.
6. É feito um *loop* (for) de 1 até n onde $y_i \in \mathbb{R}^k$ é o vetor correspondente à i -ésima linha das linhas de U .
7. Os pontos de agrupamento $(y_i)_{i=1, \dots, n}$ em \mathbb{R}^k com os algoritmos de *k-means* (No ANEXO E – K- Nearest Neighbor está o algoritmo *KNN* que é utilizado para este tipo de problema) em grupos de C_1, \dots, C_k .
8. Saída: Grupos A_1, \dots, A_k com $A_i = \{y_j | y_j \in C_i\}$.

2.2 Método de Desenvolvimento

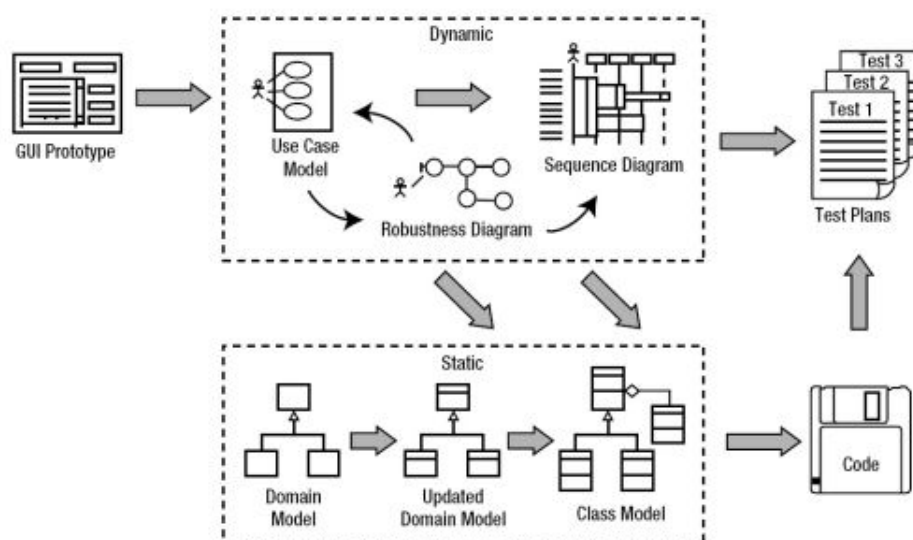
Nesta seção, é detalhado o método de desenvolvimento para a implementação da solução e do protótipo seguindo conceitos de engenharia de software e modelagem.

2.2.1 Iconix

O ICONIX (ROSENBERG; STEPHENS, 2007) é um método de desenvolver software orientado a objetos dentre os métodos ágeis composta pelas atividades Análise de Requisitos, Análise Preliminar, Projeto e Implementação. É um método simplificado que divide todas as fases de modelagem em duas visões iterativas: dinâmica e estática.

A Figura 3 é o resumo de como funciona o processo ICONIX. No processo, são utilizados os diagramas de Caso de Uso, Robustez e de Sequência na fase da visão dinâmica da modelagem e na fase da visão estática os diagramas de Domínio e de Classes.

Figura 3 – Representação do processo ICONIX



Fonte: (ROSENBERG; STEPHENS, 2007, p. 1)

A primeira atividade, Análise de Requisitos, é a fase que se envolve o início do projeto. A primeira tarefa é a conversa com o cliente, público alvo do projeto e todos interessados do projeto. Essa conversa resultará em inúmeros requisitos que darão uma estimativa de como deve ser a estruturação do projeto, visto que nesta fase, o projeto ainda não está estruturado.

O passo seguinte da análise de requisitos é entender o domínio do problema. Essa

tarefa consiste em garantir que não haja problemas de ambiguidade na compreensão do problema. O modelo de domínio (geralmente representado em *UML – Unified Modeling Language*) é usado para que fique clara a comunicação entre os membros do projeto.

Por fim, os Requisitos Comportamentais baseiam-se em cenários de como será a interação do usuário com o sistema. Começar com uma *GUI* (Interface Gráfica do Usuário – *Graphical User Interface*) é a forma de apresentar o contexto do usuário que está sendo modelado (ROSENBERG; STEPHENS, 2007).

A segunda atividade, Análise Preliminar, é uma etapa dentre a análise e o design. O diagrama de robustez descreve exatamente a atividade. O diagrama permite verificar as falhas que existem e identificar sugestões para que seu projeto não tenha possíveis ambiguidades, refinando os modelos de caso de uso da atividade anterior.

A próxima atividade, Projeto, é a construção do sistema, finalizando a visão dinâmica do sistema e início da visão estática. A preocupação no momento é a eficiência do sistema. Os principais modelos anexados a essa etapa são: Diagrama de Sequência, que fornece detalhes do que será implementado; Refinamento do Modelo de Domínio, onde devem ser inseridas as operações aos objetos de domínio e resolver os problemas reais, como o padrão de projeto que o sistema deve se adequar.

Por fim, a etapa de Implementação, é a atividade de codificação e testes com base no diagrama de Classes, formado a partir do Modelo de Domínio e Diagrama de Sequência, contendo as interações com sistemas externos.

2.3 Trabalhos Relacionados

Nesta seção serão apresentados trabalhos acadêmicos que são relacionados com os fundamentos para o desenvolvimento desta pesquisa. Os trabalhos relacionados apresentam diferentes técnicas para solucionar o roteamento de veículos.

2.3.1 Uma Metodologia para Roteamento de Veículos Escolares Utilizando Sistemas de Informação Geográfica (ROSA, 2018)

Esta dissertação de mestrado pauta-se no problema de roteamento de veículos de colégios da rede estadual na região rural do Rio de Janeiro, abordando o planejamento das rotas de uma frota de veículos que transporta alunos a partir da origem até as respectivas escolas. O problema de roteamento de ônibus escolares – *School Bus Routing*

Problem (SBRP) – é derivado do VRP. E assim como o VRP, o SBRP é um problema *NP-Difícil*. Portanto, as rotas foram obtidas através de um método meta-heurístico denominado *Adaptative Large Neighborhood Search* (ALNS) utilizando a ferramenta *VRP Spreadsheet Solver*.

Rosa 2018 apresentou a metodologia para a resolução do problema em oito fases:

1. **Definir abrangência:** a abrangência pode ser definida tanto geograficamente, ou seja, delimitar os municípios, distritos ou até colégios que serão objetivo da pesquisa, quanto os turnos que serão abordados.
2. **Geocodificar endereço das escolas, origem dos alunos e pontos de embarque:** trata-se de uma técnica de converter endereços em coordenadas geográficas – representado através de um par coordenado latitude, longitude. A geocodificação foi realizada através do *Google Maps Geocoding API*.
3. **Definir as características:** devido à delimitação de problemas reais, existem diversas classes dentro do problema do roteamento de veículos escolares que optam por abordá-los ou ignorá-los. Por exemplo, características específicas de um ambiente urbano ou rural, carregamento misto entre escolas, transporte de alunos especiais, tipo de frota entre outros. O trabalho abrangeu 15788 alunos do período noturno de 173 escolas localizadas em áreas rurais. O transporte realizado por frota heterogênea onde não é permitido carregamento misto. Visando diminuir o tempo ou distância total de viagem do veículo e levando em consideração a capacidade do veículo e o tempo máximo de percurso.
4. **Calcular a distância e tempo de percurso:** nesta fase, a distância e o tempo de percurso são calculados e inseridos em uma matriz de Origem e Destino (OD). Para estimativa do tempo dos percursos foram consideradas distâncias Euclidiana e Geodésica divididas pela velocidade média desejada. Já a distância real, que leva em consideração as vias foi obtida através do Google Maps.
5. **Montar banco de dados de georreferenciamento e da frota de veículos:** foi realizada a estruturação dos dados de localização, distância e dos veículos baseados na *VRP Spreadsheet Solver*

6. **Aplicar ferramenta para obter as rotas dos veículos:** foi aplicado uma meta-heurística denominada *Adaptive Large Neighborhood Search* utilizando o *VRP Spreadsheet Solver* para obtenção das rotas de cada veículo.
7. **Geospacilizar as rotas:** trata-se da visualização das rotas criadas através de uma interface gráfica. No trabalho foi utilizado o Google Maps como exemplo.
8. **Elaborar diagnóstico:** etapa fundamental para verificar a metodologia proposta. No caso do transporte de estudantes rurais, o diagnóstico é realizado através de parâmetros pré-estabelecidos que servem de comparação para o planejamento de ações pontuais ou globais de forma a melhorar o desempenho do sistema. De acordo com o Fundo Nacional do Desenvolvimento da Educação (FNDE), um planejamento deve considerar elementos físicos (vias, veículos), lógicos (legislação, estrutura normativa e de gestão) e os agentes (motorista e responsáveis).

Assim como no presente trabalho, o problema da dissertação também pertence à classe *OVRP*, embora esteja contida especificamente na classe de problema de roteamento de veículos escolares (*SBRP*). Outra semelhança é no modelo de planejamento das rotas que são fixas, ou seja, não são modificadas em tempo real. Entretanto, o problema é apresentado no contexto de locomoção rural do interior do Rio de Janeiro.

2.3.2 A survey of models and algorithms for optimizing shared mobility (MOURAD; PUCHINGER; CHU, 2019)

O artigo feito por (MOURAD; PUCHINGER; CHU, 2019) é uma referência importante para a resolução do VRP, os autores abordam exatamente a questão do problema emergente do aumento de número de veículos nas grandes cidades. O artigo traz resoluções para o problema de VRP em 2 aspectos: pré-organizado e em tempo-real. Pré-organizado é quando se tem informações registradas previamente, ou seja, as informações de origem e destino são inseridas antes do cálculo da rota. Já em tempo-real, todas as variáveis de cálculo são coletadas em tempo de execução, neste contexto, tornando possível calcular mudanças repentinas na rota, como fechamento de vias, ou trânsito intenso em certas vias por exemplo. O artigo também mostra que a maioria das variantes do VRP trata-se de viagens com múltiplos passageiros, uma vez que um veículo que suporta mais pessoas traz maior diminuição de veículos nas ruas. A partir da variante dos problemas, se tem,

também, uma variedade de restrições que são consideradas para cada caso do VRP, dentre elas tem se:

- Restrições de Roteamento – Onde os veículos podem ou não passar.
- Restrições de Tempo – Tempo máximo de viagem e janelas de tempo
- Restrições de Capacidade – Capacidade total do veículo
- Restrições de Custo – Custo total da viagem para os passageiros
- Restrições de Sincronia – Atendimento de demandas em tempo real

Além de avaliar o VRP em diferentes variantes do problema (diferentes veículos e demandas), há uma comparação bem ampla de resultados entre métodos heurísticos e exatos aplicados em simulação. No objetivo do projeto, que no caso se trata de resolução pré-organizada, pois atualmente não é viável dado o tempo estabelecido, trabalhar com informações em tempo-real. Assim, é uma fonte comparativa para ter uma noção de qual método se aproxima mais do ideal para o contexto do protótipo. O projeto utiliza de restrições que estão dentro da categoria de informações pré-organizadas, nós utilizaremos as seguintes restrições para alcançar seu objetivo:

- Restrições de Roteamento
- Restrições de Tempo
- Restrições de Capacidade
- Restrições de Custo

Essas restrições são o suficiente para satisfazer uma viagem pré-organizada dos usuários do transporte coletivo, se a rota atender todas essas restrições, ela será dada como uma boa rota alternativa para o usuário que deseja mais conforto e praticidade na locomoção diária.

2.3.3 Algoritmos de Solução para o Problema do Caixeiro Viajante com Passageiros e Quota (SILVA, 2017)

A dissertação de mestrado de Jean Gleisson apresenta Algoritmos de Solução para o Problema do Caixeiro Viajante com Passageiros e Quota (*PCVPQ*), sendo uma derivação no problema de roteamento de veículo. A dissertação tem como objetivo reduzir o custo total da viagem compartilhando o custo entre os passageiros que solicitam a carona, porém com o enfoque na soma mínima de quotas coletadas nas cidades visitadas pelo caixeiro. A quota é o valor do custo de deslocamento entre as cidades que será dividida igualmente entre os ocupantes, inclusive o motorista. Existem algumas restrições que devem ser respeitadas, como: O motorista é livre para decidir em quais cidades irá visitar; o custo da viagem não pode ultrapassar o valor máximo fixado pelo próprio passageiro e não é levado em consideração a duração da viagem, tempo que permanecerá nas localidades ou o número de cidades que serão visitadas durante a viagem e não haverá uma coleta bônus na cidade de origem do caixeiro. Na proposta apresentada nesta dissertação, o grafo considerado $G = (N, M, B)$, onde:

N é o conjunto de vértices que representam as cidades; M é o conjunto de arestas que representam as rodovias que ligam as cidades, onde, também, há um custo associado à aresta cujo valor é o custo de deslocamento entre as cidades; B é o bônus associado a cada cidade, que será coletado no momento que ocorre a visita da cidade.

Para a solução, o autor estudou os seguintes métodos: A heurística de carregamento ótimo (*NAIVE*) visa obter a rota de menor custo possível, e para essa rota, obter o embarque ótimo para o carro. A metaheurística *GRASP* para o *PCVPQ* descrito consiste na utilização de uma roleta para selecionar os vértices de acordo com o seu bônus e os incorporando à rota. Variações de algoritmos genéticos com mutação e meméticos com o objetivo de que as mutações aplicadas para minimizar o custo durante a execução da rota.

Como conclusão, o autor encerra dando a ênfase no problema, pois tem potencial para ser aplicado em sistemas de otimização de transporte colaborativo e os melhores desempenhos qualitativos foram alcançados pelos algoritmos *GRASP*, *MEM-SPLIT* e *MEM-SCX*, destacando-se o *GRASP* com o melhor tempo.

Assim como no trabalho mencionado, temos como objetivo o transporte coletivo de passageiros, porém levaremos em consideração:

- As cidades visitadas pelo motorista, para que haja compatibilidade de passageiros da mesma região;
- Atender um grande número de pessoas num mesmo veículo e reduzir o tempo da viagem dos passageiros e então, o custo não será levado como prioridade, pois será uma consequência;
- O tempo de viagem total, assim teremos como uma meta a viagem ser confortável para o passageiro.

Embora os objetivos e alguns métodos de resolução do problema sejam parecidos, as restrições serão diferentes, por exemplo, não teremos o sistema de coleta do bônus descrito, pois o mesmo não é relevante na nossa solução. Portanto, é importante entender que as delimitações podem gerar muitas soluções para um mesmo tipo de problema, de acordo com cada segmentação levada em consideração em cada projeto.

REFERÊNCIAS

- BECKERS, R.; DENEUBOURG, J.-L.; GOSS, S. Trails and u-turns in the selection of a path by the ant *lasius niger*. In: . [S.l.: s.n.], 1992.
- BONABEAU, E. et al. *Swarm intelligence: from natural to artificial systems*. [S.l.]: Oxford university press, 1999.
- BRANDAO, J. A tabu search algorithm for the open vehicle routing problem. *European Journal of Operational Research*, v. 157, n. 3, p. 552 – 564, 2004. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221703002388>>.
- BRASIL. *Emenda Constitucional n. 90 de 15 de setembro de 2015*. Brasília,DF: [s.n.], 2015. Disponível em: <http://www.planalto.gov.br/ccivil_03/constituicao/Emendas/Emc/emc90.htm>.
- BUENO, L. *Teoria da Complexidade Computacional*. 2011. Acessado em 27 de maio de 2019. Disponível em: <<http://professor.ufabc.edu.br/leticia.bueno/classes/aa/materiais/complexidade2.pdf>>.
- COLORNI, A. et al. Distributed optimization by ant colonies. In: CAMBRIDGE, MA. *Proceedings of the first European conference on artificial life*. [S.l.], 1992. v. 142, p. 134–142.
- CRAINIC, T. G.; GENDREAU, M.; DEJAX, P. Dynamic and stochastic models for the allocation of empty containers. *Operations research, INFORMS*, v. 41, n. 1, p. 102–126, 1993.
- DENATRAN, D. N. de T. *RESOLUÇÕES CONSOLIDADAS*. 2015. Acessado em 27 de maio de 2019. Disponível em: <<http://www.denatran.gov.br/index.php/resolucoes>>.
- DIAS, J. C. et al. An inverted ant colony optimization approach to traffic. *Engineering Applications of Artificial Intelligence*, v. 36, p. 122 – 133, 2014. ISSN 0952-1976. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0952197614001730>>.
- DIAS, R. D. Protótipo para seleção inteligente de pedidos em operações de vendas online usando heurísticas. São Paulo, p. 55, 2017.

- DORIGO, M.; BIRATTARI, M. Ant colony optimization. In: _____. *Encyclopedia of Machine Learning*. Boston, MA: Springer US, 2010. p. 36–39. ISBN 978-0-387-30164-8. Disponível em: <https://doi.org/10.1007/978-0-387-30164-8_22>.
- ESCARIO, J. B.; JIMENEZ, J. F.; GIRON-SIERRA, J. M. Ant colony extended: Experiments on the travelling salesman problem. *Expert Systems with Applications*, v. 42, n. 1, p. 390 – 410, 2015. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417414004709>>.
- FEOFILOFF, P. *Algoritmos gulosos*. 2015. Acessado em 27 de maio de 2019. Disponível em: <https://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/guloso.html>.
- FU, Z.; EGGLESE, R.; LI, L. Y. O. A new tabu search heuristic for the open vehicle routing problem. *Journal of the Operational Research Society*, Taylor & Francis, v. 56, n. 3, p. 267–274, 2005. Disponível em: <<https://doi.org/10.1057/palgrave.jors.2601817>>.
- FURUHATA, M. et al. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, v. 57, p. 28 – 46, 2013. ISSN 0191-2615. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0191261513001483>>.
- GESP, G. do Estado de S. P. *Programa Estadual de Concessões e Programa de Parcerias Público-Privadas*. São Paulo, SP: [s.n.], 2015. Acessado em: 16 de jun. de 2019. Disponível em: <<http://bit.ly/2XQsvmu>>.
- GLOVER, F. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, v. 13, n. 5, p. 533 – 549, 1986. ISSN 0305-0548. Applications of Integer Programming. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0305054886900481>>.
- GOLDBARG, M. C.; GOLDBARG, E. G.; LUNA, H. P. L. *Otimização combinatória e meta-heurísticas*. Brasil: Elsevier, 2015. 416 p. ISBN 8535278125.
- HEINEN, M. R.; OSÓRIO, F. S. Algoritmos genéticos aplicados ao problema de roteamento de veículos. *UNISINOS*, p. 7, 2006.
- INOJOSA, R. M. Redes de compromisso social. *Revista de Administração Pública*, SciELO, 1999. Disponível em: <<http://bibliotecadigital.fgv.br/ojs/index.php/rap/article/view/7628/6155>>.

- LAPORTE, G. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, v. 59, n. 3, p. 345 – 358, 1992. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/037722179290192C>>.
- LIMA, F. M. S. et al. A mixed load capacitated rural school bus routing problem with heterogeneous fleet: Algorithms for the brazilian context. *Expert systems with applications*, Elsevier, v. 56, p. 320–334, 2016.
- LOA, S. E. de Comunicação de S. *Lei N 17.021, de 27 de dezembro de 2018*. São Paulo, SP: [s.n.], 2018. Disponível em: <<http://orcamento.sf.prefeitura.sp.gov.br/orcamento/uploads/2019/lei17021.pdf>>.
- LUXBURG, U. V. A tutorial on spectral clustering. *Statistics and computing*, Springer, v. 17, n. 4, p. 395–416, 2007.
- MARINAKIS, Y.; MARINAKI, M. Honey bees mating optimization algorithm for the euclidean traveling salesman problem. *Information Sciences*, v. 181, n. 20, p. 4684 – 4698, 2011. ISSN 0020-0255. Special Issue on Interpretable Fuzzy Systems. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0020025510002938>>.
- MARINAKIS, Y.; MARINAKI, M.; DOUNIAS, G. Honey bees mating optimization algorithm for large scale vehicle routing problems. *Natural Computing*, v. 9, n. 1, p. 5 – 27, Mar 2010. ISSN 1572-9796. Disponível em: <<https://doi.org/10.1007/s11047-009-9136-x>>.
- MARTINS, W. T. *Índice de Avaliação da Qualidade do Transporte Público por Ônibus a Partir da Definição de Serviço Adequado*. 117 p. Dissertação (Mestrado) — Universidade de Brasília, UnB - Brasília, DF, 70910-900, 2015.
- METRO, C. do Metropolitano de S. P. *Para que servem as informações da pesquisa Origem e Destino*. São Paulo, SP: [s.n.], 2012. Acessado em: 16 de jun. de 2019. Disponível em: <<http://www.metro.sp.gov.br/pesquisa-od/informacoes-od.aspx>>.
- MOURAD, A.; PUCHINGER, J.; CHU, C. A survey of models and algorithms for optimizing shared mobility. *Transportation Research Part B*:

Methodological, v. 123, p. 323 – 346, 2019. ISSN 0191-2615. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0191261518304776>>.

NOGUEIRA, D. K. Introdução à teoria dos grafos: proposta para o ensino médio. 2015.

OLIVEIRA, A. F. M. d. A. *Extensões do problema do caixeiro viajante*. Dissertação (Mestrado) — Universidade de Coimbra, 2015.

ONGSAKUL, W.; PETCHARAKS, N. Unit commitment by enhanced adaptive lagrangian relaxation. *IEEE Transactions on Power Systems*, IEEE, v. 19, n. 1, p. 620–628, 2004.

PANCHER, A. M.; FREITAS, M. I. C. de. *Georreferenciamento*. São Paulo, SP: [s.n.], 2018. Acessado em: 16 de jun. de 2019. Disponível em: <<https://bit.ly/2IStcFy>>.

PAPADIMITRIOU, C. H.; STEIGLITZ, K. *Combinatorial Optimization: Algorithms and Complexity (Dover Books on Computer Science)*. Estados Unidos: Dover Publications, 1998. 496 p. (Dover Books on Computer Science). ISBN 0486402584.

REDE NOSSA SAO PAULO. *9 Pesquisa Sobre Mobilidade Urbana: Semana da Mobilidade 2015*. 2015. 58 p. Acessado em: 08 de abr. de 2019. Disponível em: <<https://www.mobilize.org.br/midias/pesquisas/pesquisa-sobre-mobilidade-urbana—rede-nossa-sp.pdf>>.

REDE NOSSA SAO PAULO, IBOPE. *Viver em São Paulo, Mobilidade Urbana na Cidade*. 2018. 37 p. Acessado em: 08 de abr. de 2019. Disponível em: <<https://bit.ly/2NklPfm>>.

RNSP, R. N. S. P. *Debates sobre mobilidade e transporte*. São Paulo, SP: [s.n.], 2018. Acessado em: 16 de jun. de 2019. Disponível em: <<https://www.nossasaopaulo.org.br/campanhas/#20>>.

ROSA, B. A. B. *Uma Metodologia para Roteamento de Veículos Escolares Utilizando Sistemas de Informação Geográfica*. 99 p. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio de Janeiro - PUCSP, Rio de Janeiro, 2018.

ROSENBERG, D.; STEPHENS, M. *Use Case Driven Object Modeling with UML Theory and Practice*. [S.l.]: Apress, 2007. 438 p.

SHI, J.; MALIK, J. Normalized cuts and image segmentation. *Departmental Papers (CIS)*, p. 107, 2000.

SILVA, J. G. d. S. *Algoritmos de solução para o problema do caixeiro viajante com passageiros e quota*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2017.

SPTRANS. *Política tarifária e seus efeitos no sistema de transporte*. São Paulo, SP: [s.n.], 2018.

SPTRANS. *Área do Desenvolvedor*. São Paulo, SP: [s.n.], 2019. Acessado em: 16 de jun. de 2019. Disponível em: <<http://www.sptrans.com.br/desenvolvedores/>>.

STEINER, M. T. A. et al. O problema de roteamento no transporte escolar. *Pesquisa Operacional*, SciELO Brasil, v. 20, n. 1, p. 83–99, 2000.

TAILLARD, E. Robust taboo search for the quadratic assignment problem. *Parallel Computing*, v. 17, n. 4, p. 443 – 455, 1991. ISSN 0167-8191. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167819105801474>>.

TOTH, P.; VIGO, D. The vehicle routing problem. Estados Unidos, p. 17, 2002.

TSUBAKITANI, S.; EVANS, J. R. An empirical study of a new metaheuristic for the traveling salesman problem. *European Journal of Operational Research*, v. 104, n. 1, p. 113 – 128, 1998. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221796003347>>.

UBER. *Movement Cities*. 2019. Acessado em: 16 de jun. de 2019. Disponível em: <<https://movement.uber.com/cities>>.

ZIVIANI, N. *Projeto De Algoritmos Com Implementações Em Pascal E C*. Brasil: Cengage Learning, 2010. 660 p. ISBN 9788522110506.

ANEXO

Anexo A – Tabela de Indicadores de Qualidade do Serviço

Tabela 3 – Indicadores do serviço adequado.

PARÂMETRO DO SERVIÇO ADEQUADO	DEFINIÇÃO DO PARÂMETRO	INDICADOR EQUIVALENTE	SIGLA	DEFINIÇÃO ADOTADA NO INDICADOR
Regularidade	Prestação do serviço conforme contrato, sem interrupções	Pontualidade	P_N	Oferta do serviço de forma contínua e conforme planejado
Continuidade	Manutenção da oferta em caráter permanente	Tempo	T_E	Tempo de realização da viagem, incluindo transbordos
Eficiência	Atendimento ao usuário com o melhor uso dos recursos	Lotação	L_O	Oferta de lugares e conforto das viagens
Segurança	Confiança oferecida ao usuário na prestação do serviço	Segurança	S_G	Serviço seguro, sem incidência de acidentes e assaltos
Atualidade	Modernidade de técnicas, equipamentos e instalações	Informações	I_F	Uso de técnicas para fornecer informação ao usuário, sem necessidade de solicitação
Generalidade	Serviço com atendimento a todos os usuários, sem distinções	Veículos	V_E	Acesso de pessoas com necessidades especiais e conservação dos veículos
Cortesia	Serviço prestado com urbanidade ao usuário	Tripulação	T_R	Comportamento dos operadores e sua relação com usuários
Modicidade das tarifas	Tarifas compatíveis com a renda do usuário	Tarifa	T_A	Representatividade da tarifa na renda do usuário

Fonte: (MARTINS, 2015, p. 43)

Anexo B – Métodos Exatos Utilizados no VRP

Branch-and-bound

Esta estratégia consiste em solucionar problemas de otimização com restrições de variáveis inteiras. A partir de um problema inicial, é criada uma série de ramificações (*Branch*) das restrições (*Bound*) que existem neste problema, de modo que, as restrições não permitam uma solução parcial do problema. Esta estratégia monta uma árvore de decisão com base em limites superiores e inferiores, impedindo que algumas soluções ruins sejam exploradas. Este método pode ser utilizado com outros métodos de programação linear, como por exemplo, o SIMPLEX (LAPORTE, 1992). O *Branch-and-Bound* elimina ramificações de restrições caso se encontre uma das condições:

- 1.O problema restrito é impossível.
- 2.A solução ótima do ramo é inteira.
- 3.A solução encontrada no ramo é pior do que a melhor solução atual.

Branch-and-cut

Esta estratégia é uma melhoria do algoritmo *Branch-and-bound*. Nela, ocorre a resolução dos problemas relaxados (sem restrições de integralidade) do *Branch-and-bound*, utilizando um plano de cortes. O algoritmo passa nó a nó da árvore, executando o plano de cortes e checando se naquele nó, a solução ótima do subproblema obedece às restrições de integralidade ou se o limitante inferior é maior que a solução já conhecida, caso uma dessas situações ocorra, o nó é descartado. Caso não seja encontrada uma inequação válida para o problema de separação, aplica-se a ramificação.

Branch-and-price

Em casos de problemas de otimização com variáveis integrais, onde se considera todas as variáveis do problema simultaneamente, torna-se inviável a utilização das estratégias anteriores. Visto esse tipo de problema, foi criada uma técnica chamada de

Geração de Colunas, com intuito de resolver esse tipo de problema. A técnica visa selecionar variáveis com o maior potencial para otimizar o problema. A estratégia *Branch-and-price* é uma versão da estratégia *Branch-and-bound*, que utiliza a Geração de Colunas em cada nó da árvore de decisão.

Anexo C – Métodos Heurísticos Utilizados no VRP

Clarke e Wright

Esta foi uma estratégia extremamente importante para o desenvolvimento dos algoritmos voltados ao VRP, este algoritmo serviu como base para a construção de outros algoritmos sofisticados de roteamento de rotas e veículos (HEINEN; OSÓRIO, 2006). O algoritmo cria grafos contendo os clientes e possíveis rotas, já o depósito é lido a partir de alguma estrutura de armazenamento, então, são calculadas $n-1$ rotas de um cliente ao depósito onde n é o número de nós do grafo. A partir destes pares de nós, é calculado o custo da rota a partir da fórmula (ROSA, 2018, p. 32):

$$s_{ij} = c_{i0} + c_{0j} - c_{ij}$$

Onde s_{ij} representa a economia com relação aos nós i e j , e c_{ij} representa o custo de se ir do nó i ao nó j . As economias são colocadas, em ordem decrescente, em uma tabela após o cálculo.

O algoritmo de *Clark e Wright* é resolvido em tempo polinomial, pois, a complexidade é da ordem $\mathcal{O}(n^2)$. Há, no entanto, uma desvantagem de se usar a estratégia de *Clark e Wright*, pois o algoritmo descartará todos os nós que não se encontram nos extremos das rotas, ou seja, os nós internos não entram na tabela de economia.

Mole e Jameson

A estratégia de *Mole e Jameson* é uma melhoria do algoritmo de *Clark e Wright*, ele tem como um ponto forte, considerar os nós internos que são descartados no teste de economia do algoritmo de *Clark e Wright*. O algoritmo começa carregando o grafo a partir de uma matriz de adjacências e de uma lista de nós livres (nós que não pertencem à nenhuma rota). Posteriormente, é construído um grafo que não contém o nó do depósito a partir da matriz de adjacências e da lista de nós livres. A partir de algum critério pré-definido, um par de nós é escolhido para ser a rota inicial. Após definida a rota inicial, um laço começará a inserir os nós na rota, seguindo dois critérios: O de **Proximidade** e o de **Economia**. Para realizar o cálculo da **Proximidade** dos nós é utilizada a seguinte fórmula (ROSA, 2018, p. 33):

$$\alpha(i, k, j) = c_{ik} + c_{kj} - \lambda c_{ij}$$

Onde c_{ik} representa o custo entre os nós i e k , c_{kj} representa o custo entre os nós k e j . Se nenhuma violação for quebrada, o nó com menor distância será inserido na rota. O cálculo da economia é aquele que apontará qual o melhor lugar para o nó ser inserido, este cálculo é dado por:

$$\beta(i, k, j) = \mu_0 k - \alpha(i, k, j)$$

Onde $\mu_0 k$ representa a distância entre o depósito e o nó k . O local selecionado representa o que tem mais economia.

Anexo D – Métodos Meta-Heurísticos Utilizados no VRP

Busca Tabu

A etimologia da palavra *tabu* tem origem do *Tongan*, idioma falado em Tonga, na Polinésia e indica algo proibido por ser sagrado e que não pode ser tocado ou mencionado. O algoritmo da busca tabu é assim nomeado por ser composto de estratégias que permitem *proibir* determinadas ações durante a varredura de vizinhanças e a escolha de soluções.

A busca tabu foi introduzida na literatura por Fred Glover em 1986 (GLOVER, 1986) e posteriormente aperfeiçoada em outros trabalhos acadêmicos. É um método que permite tanto uma abordagem probabilística (CRAINIC; GENDREAU; DEJAX, 1993) quanto determinística (ONGSAKUL; PETCHARAKS, 2004). Ao contrário de muitos métodos meta-heurísticos, a busca tabu não possui analogia com fenômenos naturais. Os princípios de uma busca tabu segundo Golbarg *et al.* (2015) são:

1. Uma busca eficiente não deve revisitar soluções;
2. Manter registro de todas as soluções visitadas é computacionalmente dispendioso. É mais eficiente armazenar alterações na solução ou em variáveis; e
3. Memorizar alterações nas soluções não elimina a possibilidade de reexame de soluções, apenas diminui a chance dessa possibilidade.

Um dos princípios da busca tabu é aproveitar-se de um esquema de gestão de memória auxiliar de curto, médio e longo prazo para evitar examinar mais de uma vez uma mesma configuração em uma varredura em busca de uma solução. Este conjunto de proibições – usualmente denominado *condições tabu* – é registrado em memória dinâmica, permitindo que seja alterado de acordo com o progresso da busca, visto que podem ocorrer evidências de que uma configuração de solução não está sendo considerada e estas proibições precisam ser desativadas. Além de evitar repetições, uma otimização das configurações da busca permite aumentar a chance da busca escapar da atração de mínimos locais.

Fundamentalmente, algoritmos de busca tabu são orientados à varredura de vizinhanças específicas (denominada *intensificação*) com configuração de foco de busca

variável (denominada *diversificação*). Um algoritmo de busca tabu visa crucialmente minimizar a quantidade de configurações visitadas e fornecer condições de escape de mínimos locais. Portanto, para atingir este objetivo, uma busca tabu é composta por:

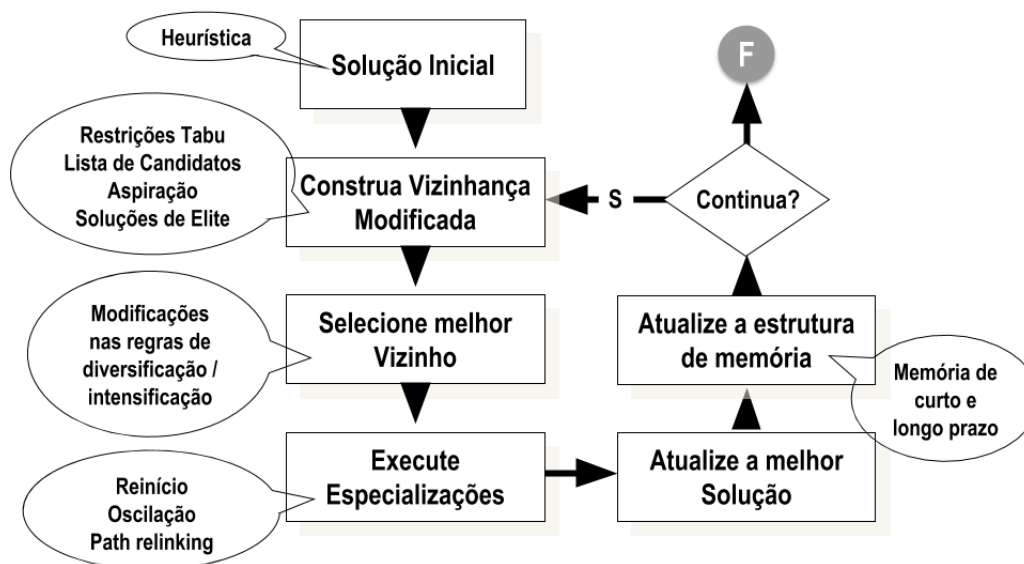
- Memória de curto prazo para preservar o histórico imediato da busca;
- Estratégias de memória de médio e longo prazo com o objetivo de equilibrar o esforço computacional da *intensificação* com o utilizado na *diversificação*;
- Regras que permitam ajustar gestão da memória, critérios de perturbação e de escolha da configuração.

A busca tabu armazena na memória algumas soluções e armazena as alterações posteriores nas próprias soluções. Estas modificações nas soluções são denominadas de *perturbações*. Uma *perturbação* é uma alteração no valor ou na composição das variáveis de uma solução. Goldbarg *et al.* (2015) exemplificam alguns exemplos de possíveis movimentos em uma configuração:

- Mudar as atribuições de uma variável binária de zero para um, ou vice-versa.
- Permutar os vértices em uma solução de rota.
- Introduzir ou retirar arcos em uma grafo solução.
- Alterar as atribuições realizadas sobre os vértices de um grafo solução.

A figura 4 representa através de um fluxograma o passo a passo de um algoritmo de Busca Tabu. Os comentários identificam o propósito e o nome de cada passo.

Figura 4 – Passo a passo com comentários de um algoritmo de Busca Tabu.



Fonte: (GOLDBARG; GOLDBARG; LUNA, 2015, p. 97)

O Algoritmo 1 representa em pseudocódigo, um algoritmo de busca tabu. A variável p_{max} controla o máximo de iterações. A variável q_{max} controla as iterações sem melhoria. $N(s)$ simboliza a vizinhança da configuração s e $f(s)$ representa o valor da configuração s .

Colônia de Formigas e Enxame de Abelhas

Conforme dito anteriormente, muitos métodos meta-heurísticos são inspirados em fenômenos químicos, biológicos, sociais e naturais, como é o caso da Colônia de Formigas e Enxame de Abelhas. Algoritmos em Colônia de Formigas e Enxame de Abelhas são pautados na capacidade de como organismos simples com conceitos modestos de inteligência – geralmente associados a insetos ou animais de cognição simples – são capazes de atuar socialmente em conjunto e criar uma organização robusta.

Esta forma de organização social com razoável grau de autonomia é denominado de *Inteligência Coletiva* e um dos precursores do estudo foi William Morton Wheeler, em 1911. Esta forma de inteligência é tão extraordinária que o Prêmio Nobel de 1911, Maurice Maeterlinck, já questionava em seu livro *La Vie des Abeilles* e vários estudos

Algoritmo 1 Pseudocódigo da busca tabu (GOLDBARG; GOLDBARG; LUNA, 2015, p. 89)

```

1: gerar uma solução inicial  $s$ , e fazer  $s^* := s$ 
2: inicializar a Lista Tabu ( $T$ )
3: inicializar contadores  $p$  e  $q$ 
4: enquanto  $p \neq p_{\max}$  e  $q \neq q_{\max}$  faça
5:   selecionar o melhor vizinho  $s' \in N(s) \setminus T$ 
6:   selecionar o melhor vizinho  $s'' \in N(s) \cap T$ 
7:
8:   se  $f(s'') < f(s')$  e  $f(s'') < f(s^*)$  então
9:      $s' \leftarrow s''$  ▷ Aspiração
10:  fim se
11:
12:  se  $f(s') < f(s^*)$  então
13:     $s^* \leftarrow s'$ 
14:     $q \leftarrow 0$ 
15:  fim se
16:
17:  se  $f(s') < f(s)$  então
18:    colocar o movimento inverso  $(s', s)$  na lista  $T$ 
19:    atualizar  $T$ 
20:  fim se
21:   $s \leftarrow s'$ 
22:   $p \leftarrow p + 1$ 
23:   $q \leftarrow q + 1$ 
24: fim enquanto
25: Devolver a solução  $s^*$ 

```

posteriores foram realizados com os mais diversos tipos de insetos (DORIGO; BIRATTARI, 2010). A definição segundo Bonabeau *et al.* 1999:

Inteligência Coletiva é uma abordagem que permite o desenvolvimento de algoritmos de processamento distribuído inspirados no comportamento de insetos sociais ou de outras sociedades animais.

(BONABEAU *et al.*, 1999, p. 7)

Na área de Otimização Combinatória, os precursores na utilização da Colônia de Formigas foram Beckers *et al.* 1992, onde foi demonstrado através de um experimento que formigas são capazes de encontrar o caminho mais curto entre o formigueiro e uma fonte de alimentação explorando coletivamente o feromônio que as formigas deixam no chão enquanto caminha; e Coloni *et al.* 1992, que ainda em 1992 publicaram estudo sobre otimização distribuída baseado em Colônias de Formigas.

Sob o ponto de vista sistêmico, estes algoritmos trabalham através de processos de auto-organização. A auto-organização pode ser atribuída a um sistema aberto – um sistema que permite troca de energia com o ambiente – que contém a propriedade de conseguir crescer em complexidade de forma organizada sem que exista um guiamento centralizado ou gestão externa ao sistema.

Os sistemas auto-organizados emergem padrões. Um padrão é um arranjo privado e organizado de objetos que é preservado no espaço e no tempo. Quando emergentes, estes arranjos permitem que as propriedades do sistema se alterem de forma organizada.

O processo que se opõe à desorganização (*entropia*) é denominado de *negentropia* ou *sintropia*. O Nobel da Física Erwin Schrödinger, distinguiu sistemas com tendências entrópicas e outros que possuem dispositivos negentrópicos e que são capazes de se organizar em níveis cada vez mais complexos e se auto-organizar e regenerar.

Os algoritmos de Colônia de Formigas são compostos de agentes (*formigas*) que cooperam globalmente para encontrar uma solução. À medida que as soluções são construídas, estruturas de vizinhança ou especiais são marcadas como forma de comunicação. Essa comunicação é empregada assim como a utilização de feromônios na organização das formigas para a construção de uma solução aos passos da formiga que está procurando alimento. Desta forma, a construção da solução percorre uma trilha de decisões marcadas anteriormente.

Algoritmos de Colônia de Formigas seguem algumas características importantes:

- As formigas não se comunicam diretamente;
- Cada formiga constrói uma solução movendo-se em uma sequência finita de estados.
- Estes movimentos são feitos de acordo com probabilidade que leva em consideração informações da própria formiga (memória simples de ações locais) e da trilha de feromônio (ações globais)

ANEXO E – K- Nearest Neighbor

Para realizar o agrupamento de candidatos a serem integrados numa possível rota, a partir do ponto de origem do motorista, é possível utilizar um algoritmo de *Supervised Machine-Learning* chamado *K-Nearest Neighbour* (KNN). O algoritmo tem o seguinte funcionamento:

A ideia principal do algoritmo é resolver problemas de classificação e regressão. Por ser supervisionado, ele necessita de interação humana no recebimento de entradas e retroalimentação de dados para melhorar seu desempenho. O KNN faz o agrupamento dos dados tendo o princípio inicial de que dados semelhantes têm maior proximidade (funções de distância euclidiana), tendo isso, a classificação ocorre de modo que K é o número de dados a serem avaliados para o agrupamento do KNN. Por ser um algoritmo que requer retroalimentação de dados, é necessário várias execuções para se atingir um valor de K que seja adequado ao problema.

Colocando o KNN no contexto do projeto, é possível realizar a classificação por proximidade a partir de pontos geográficos do mapa, que no caso, seriam a semelhança entre os usuários. Após a comparação dos dados é definido a distância entre eles, e assim, o KNN vai classificando-os e agrupando-os por regiões de origem e destino semelhantes.