



Universidade Federal da Paraíba
Centro de Tecnologia
Departamento de Engenharia de Produção
Curso de Graduação em Engenharia de Produção Mecânica

JOÃO VÍTOR DE ARAÚJO GOMES

**Resolução do problema de determinação de rotas para o transporte universitário
de Timbaúba-PE com destino à João Pessoa-PB.**

JOÃO PESSOA

2024

JOÃO VÍTOR DE ARAÚJO GOMES

**Resolução do problema de determinação de rotas para o transporte universitário
de Timbaúba-PE com destino à João Pessoa-PB.**

Trabalho de Conclusão de Curso de Engenharia de Produção Mecânica do Centro de Tecnologia da Universidade Federal da Paraíba, apresentado como requisito à obtenção do título de Bacharel em Engenharia de Produção Mecânica.

Orientadora: Profa. Me. Alessandra Berenguer de Moraes.

JOÃO PESSOA

2024

Catálogo na publicação
Seção de Catalogação e Classificação

G633r Gomes, Joao Vítor de Araújo.

Resolução do problema de determinação de rotas para o transporte universitário de Timbaúba-PE com destino à João Pessoa-PB. / Joao Vítor de Araújo Gomes. - João Pessoa, 2024.

59 f. : il.

Orientação: Alessandra Berenguer de Moraes.
TCC (Graduação) - UFPB/CT.

1. Pesquisa Operacional. 2. Problema do caixeiro viajante. I. Moraes, Alessandra Berenguer de. II. Título.

UFPB/CT/BSCT

CDU 658.5:621(043.2)




UNIVERSIDADE FEDERAL DA PARAÍBA – UFPB
CENTRO DE TECNOLOGIA – CT
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO – DEP

FOLHA DE APROVAÇÃO


Aluno: JOÃO VÍTOR DE ARAÚJO GOMES

Título do trabalho: **Resolução do problema de determinação da rota ótima para o transporte universitário de Timbaúba-PE à João Pessoa-PB.**


Trabalho de Conclusão de Curso defendido e aprovado em 03 de maio de 2024 pela banca examinadora:

Documento assinado digitalmente
 ALESSANDRA BERENQUER DE MORAES
Data: 11/05/2024 19:24:30-0300
Verifique em <https://validar.iti.gov.br>

Prof.^a Ma. ALESSANDRA BERENQUER DE MORAES – Orientadora – DEP/CT/UFPB

Documento assinado digitalmente
 LIANE MARCIA FREITAS E SILVA
Data: 13/05/2024 10:49:46-0300
Verifique em <https://validar.iti.gov.br>

Prof.^a Dra. LIANE MÁRCIA FREITAS E SILVA – Membro – DEP/CT/UFPB

Documento assinado digitalmente
 LUCIANO CARLOS AZEVEDO DA COSTA
Data: 13/05/2024 10:53:09-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. LUCIANO CARLOS AZEVEDO DA COSTA – Membro – DEP/CT/UFPB

DEDICATÓRIA

Em homenagem a todos os meus queridos e adoráveis professores desde o meu primeiro dia letivo escolar até aqui, vocês fizeram parte da minha construção como cidadão e ser social.

Acredito que ser professor é ser também revolucionário. É preciso ter muita vontade de repassar o conhecimento e de vencer as adversidades de um Brasil tão promiscuo, ignorante..., mas ao mesmo tempo repleto de cultura e de pessoas aguerridas que sorriem em meio ao caos social.

Aqui também incluo meu pai, João Marcelo Gomes Ferreira que além de pai é também um grande lecionador de história, pós-graduado, que me ensina todos os dias o quanto é importante valorizar os fatos históricos para entendermos quem somos e que podemos escolher sobre o que seremos.

A educação é a fonte de transformação, não apenas da construção social, mas da consciência humana.

A vocês, dedico não só este documento, dedico também todo meu conhecimento.

AGRADECIMENTOS

A Mabel Vieira de Araújo, mulher, do lar, empreendedora, boleira, gentil, amorosa, amada, graciosa, educadíssima, sincera, correta, direta... minha querida mãe tem tantas qualidades que não caberiam nas sessenta e três páginas deste trabalho! Me questiono: Será que a palavra gratidão explica meu sentimento neste momento?! Mas consigo a resposta dela mesma que me ensinou sobre o agradecer que é sinônimo de cuidar que, por conseguinte, é sinônimos de amor. Mãe, obrigado por tanto amor, cuidado, carinho e atenção durante esses dez anos de graduação.

A João Marcelo Gomes Ferreira, meu pai o qual já foi anteriormente citado, mas preciso agradecer a persistência para que eu seguisse, para que eu não desistisse. Que foi financeiramente muito importante durante minha graduação, que não me deixou desamparado em momento algum, que buscou mesmo em meio as nossas diferenças, entender os meus anseios e me orientar para a vida, para não apenas ser cidadão, mas praticar a cidadania. Exemplo político, racional e ideológico. Obrigado, me sinto orgulhoso, meu pai!

A Marcella Araújo Gomes e Mariana Araújo Gomes, minhas maravilhosas irmãs. Elas que são fonte de inspiração e espelho em minha vida. Que são um terço de mim, cada, que compartilham comigo uma linhagem, não apenas isto, compartilham comigo momentos indescritíveis, únicos e sentimentos que se renovam a cada reencontro. Obrigado por terem suportado meus defeitos, emitido críticas construtivas para meu sucesso e me amado cada uma a sua maneira. Estarei com vocês onde quer que estejam.

A minha célebre amada noiva, namorada e mulher Luana Augusta Gonçalves de Oliveira, que tanto me impulsiona, me inspira, me supervaloriza, me edifica e que tanto me oferta amor dos mais diversos jeitos e formas inimagináveis. Lu, saiba que você foi fundamental para que este trabalho fosse concluído, com sua paciência e compreensão, com seu carinho e sua atenção. Obrigado por esse amor tão generoso. Estendo a gratidão aos meus sogros Luciana Pantaleão e José Eudes e meu cunhado Lucas Josuel por serem também meu segundo lar e me acolherem com tanto carinho.

A minhas tias Eleonora Araújo e Crisálida Araújo por serem como mães, cheias de carinho, apoio, sempre atenciosas e com as mãos sempre estendidas para me ajudarem. Amo vocês demais.

Ao meu grande tio Rômulo Xavier e sua filha (minha prima) Vitória de Araújo Xavier que dedicaram seu tempo me ensinando práticas e técnicas de programação em VBA e Python, fundamentais para construção dos algoritmos aqui apresentados. Vocês não imaginam o quanto sou feliz por tê-los em minha vida, obrigado!

A Túlio Venâncio, meu grande amigo do curso de graduação, que compartilhamos tantos momentos durante o curso e que foi fundamental para que eu alcançasse meu diploma, com sua paciência, compartilhamento e compreensão. Aos amigos de curso Edvaldo, Charles, Karol e Jucélia por também terem contribuído diretamente com minha formação. A Thaís Machado por ter dedicado um tempinho para me auxiliar com as normas ABNT do trabalho, com sua paciência e dedicação genuína. A minha amiga

Rebeca Macedo, amiga de longa data, engenheira de software e ao amigo Matheus Silva também da graduação, que me ajudaram diretamente na construção dos códigos em Python pelas heurísticas do vizinho mais próximo e código genético, respectivamente, esse trabalho também é parte de vocês dois, todo mérito, honra e respeito a ambos.

A Jonathan, Lindoaldo e Teófilo, amigos de vivência em João Pessoa-PB, dividimos casa, comida, lazer, emoções e aprendizados suficientes que ficarão marcados para sempre na minha história, vocês são grandes pessoas de inúmeros valores que apenas foram reafirmados com nosso dia a dia. Também agradeço imensamente a Anderson Vitor, Eduardo Albuquerque, Luiz Eduardo, Laura Verbena e Cristian Marinho por serem amigos fonte de inspiração para o sucesso como profissional e ser humano, são irmãos para a vida!

Aos meus demais amigos que são parte fundamental da minha história e que com eles aprendo todos os dias coisas que acrescentam na vida.

A minha querida orientadora Prof.^a Alessandra Berenguer, não há palavras para descrever tamanho respeito e admiração pela sra.

A empresa Rosa Master, que me compreendeu no momento final de graduação e deu carta branca para que desse continuidade ao curso, me dando insumos necessários para que chegasse até aqui. A todos meus amigos e colegas de trabalho por serem parte fundamental do meu dia a dia e contribuintes do meu crescimento profissional.

A todos que puderam de alguma forma, do seu jeito, contribuir com minha vida. Espero demonstrar minha gratidão não apenas em palavras, mas também com ações positivas que reafirmem todos os dias os nossos laços.

Enfim, que este trabalho seja fundamental na disseminação de conhecimento para quem o ler. É uma enorme satisfação poder contribuir com a comunidade acadêmica e ser ponte para o aprendizado.

EPÍGRAFE

“... que é muito difícil você vencer a injustiça secular, que dilacera o Brasil em dois países distintos: o país dos privilegiados e o país dos despossuídos.”

Ariano Suassuna

RESUMO

A Prefeitura Municipal de Timbaúba (PMT), visando entregar uma maior acessibilidade e oportunidades aos seus munícipes, disponibiliza transportes públicos diários com destino às universidades e faculdades da Mata Norte do Estado e nas capitais de Pernambuco e Paraíba oferecendo uma oportunidade logística de estudos ao seu munícipe. No município de Timbaúba-PE, localizado à 88km de João Pessoa-PB, não é diferente, a demanda diária está em torno de cem discentes entre instituições públicas e privadas sendo a metade apenas na Universidade Federal da Paraíba (UFPB). A ideia proposta é a aplicabilidade do problema do caixeiro viajante para entender como otimizar ou validar as rotas e conseqüentemente o tempo de entrega desses discentes em suas respectivas instituições sem onerar mais despesas com os transportes públicos e que estão alocados especificamente para esta atividade cotidiana. Neste problema é tratado um estudo de natureza aplicada, objetivos explanatórios e normativos e abordagem quantitativa para procedimentos do estudo de caso com a utilização do complemento solver da ferramenta Excel (R) disponibilizada pela empresa Microsoft e aplicabilidade da linguagem de programação Python na heurística do vizinho mais próximo e na modelagem de programação linear MTZ (1960), todos às teorias de Pesquisa Operacional para comprovar matematicamente a otimização do objetivo da função a ser determinada. O trabalho obteve uma confirmação da rota usual, indicando, portanto a boa utilização do tempo de viagem, validando as informações coletadas e também sugerindo que o percurso inverso mantém-se com mesma economia de tempo de percurso entregando os alunos em tempo hábil em seus respectivos horários de aula.

Palavras-chave: Problema do Caixeiro Viajante; Linguagem Python; Logística; Pesquisa Operacional.

ABSTRACT

The Municipality of Timbaúba (PMT), aiming to provide greater accessibility and opportunities to its citizens, provides daily public transport to universities and colleges in Mata Norte of the State and in the capitals of Pernambuco and Paraíba, offering a logistical opportunity for studies at your disposal. citizen. In the municipality of Timbaúba-PE, located 88km from João Pessoa-PB, it is no different, the daily demand is around one hundred students between public and private institutions, half of which are only at the Federal University of Paraíba (UFPB). The proposed idea is the applicability of the traveling salesman problem to understand how to optimize or validate the routes and consequently the delivery time of these students to their respective institutions without incurring more expenses with public transport, which are allocated specifically for this daily activity. This problem deals with a study of an applied nature, explanatory and normative objectives and a quantitative approach to case study procedures using the solver complement of the Excel (R) tool made available by Microsoft and the applicability of the Python programming language in neighbor heuristics. closest and in linear programming modeling MTZ (1960), all to Operations Research theories to mathematically prove the optimization of the objective of the function to be determined. The work obtained confirmation of the usual route, therefore indicating good use of travel time, validating the information collected and also suggesting that the reverse route remains with the same savings in travel time, delivering students in a timely manner to their respective class schedules.

Keywords: Traveling Salesman Problem; Python language; Logistics; Operational Research.

LISTA DE FIGURAS

FIGURA 1 - MODULAÇÃO DE PROBLEMAS PARA PESQUISA OPERACIONAL

FIGURA 2: FORMULAÇÃO MATEMÁTICA DO CAIXEIRO VIAJANTE

FIGURA 3: GRAFO DA ROTA DE IDA COMUM REALIZADA

FIGURA 4: GRAFO DA ROTA DE RETORNO COMUM REALIZADA

FIGURA 5: MODULAÇÃO DFJ (1954) COM ALTERNATIVA DE ELIMINAÇÃO DE SUBROTAS ATRAVÉS DO MÉTODO SIMPLEX.

FIGURA 6: GRAFO DA ROTA DE IDA IDEAL ENCONTRADA PELO SOLVER.

LISTA DE TABELAS

TABELA 1 – CRONOMETRAGEM (BR101)

TABELA 2 – CRONOMETRAGEM (BR230)

TABELA 3 – TEMPO DE VIAGEM PELO GOOGLE MAPS (BR101)

TABELA 4 – TEMPO DE VIAGEM PELO GOOGLE MAPS (BR230)

TABELA 5 – CONFIGURAÇÃO DOS LOCAIS

TABELA 6 – ROTA BR230 COM REFERÊNCIA DO GOOGLE MAPS

TABELA 7 – ROTA BR230 COM CRONOMETRAGEM DO PERCURSO

TABELA 8 – ROTA BR101 COM REFERÊNCIA DO GOOGLE MAPS

TABELA 9 – ROTA BR101 COM CRONOMETRAGEM DO PERCURSO

TABELA 10 – MÉTODOS UTILIZADOS E SOLUÇÕES OBTIDAS

TABELA 11 – HORÁRIO DE DESEMBARQUE ATRAVÉS DA ROTA COMUM

TABELA 12 – HORÁRIO DE DESEMBARQUE ATRAVÉS DA ROTA MENSURADA

TABELA 13 – HORÁRIO DE EMBARQUE PARA RETORNO ATRAVÉS DA ROTA COMUM.

TABELA 14 – HORÁRIO DE EMBARQUE PARA RETORNO ATRAVÉS DA ROTA MENSURADA.

LISTA DE ABREVIATURAS E SIGLAS

AGs - Algoritmos Genéticos

UNIPÊ - Centro Universitário de João Pessoa

UNINASSAU - Centro Universitário Maurício de Nassau

COLAB - Colaboratory

DFJ - Dantzig, Fulkerson, Johnson

FPB - Faculdade da Paraíba

FTS - Faculdade Três Marias

MEC - Ministério da Educação do Brasil

MTZ - Miller, Tucker, Zemlin

ORE 3 – Ônibus Rural Escolar 3

PB - Paraíba

PE - Pernambuco

PMT – Prefeitura Municipal de Timbaúba

PO - Pesquisa Operacional

PCV - Problema do Caixeiro Viajante

SET - Secretaria de Educação de Timbaúba

UFPB - Universidade Federal da Paraíba

UF - Universidades Federais

SUMÁRIO

| | |
|--|-----------|
| 1. INTRODUÇÃO | 14 |
| 1.1. JUSTIFICATIVA | 15 |
| 1.2. OBJETIVOS | 15 |
| 1.2.1. OBJETIVO GERAL..... | 15 |
| 1.2.2. OBJETIVOS ESPECÍFICOS..... | 16 |
| 2. FUNDAMENTAÇÃO TEÓRICA..... | 16 |
| 2.1. PESQUISA OPERACIONAL..... | 16 |
| 2.2. PROGRAMAÇÃO LINEAR | 19 |
| 2.3. MÉTODOS HEURÍSTICOS | 22 |
| 2.4. CAIXEIRO VIAJANTE | 23 |
| 2.5. ALGUNS CONCEITOS DE LOGÍSTICA..... | 26 |
| 2.6. PROGRAMAÇÃO: PEQUENA ABORDAGEM SOBRE A LINGUAGEM PYTHON 28 | |
| 3 METODOLOGIA..... | 30 |
| 3.1. FASE 1 – O PROBLEMA: SITUAÇÃO ATUAL DA ROTA DO TRANSPORTE UNIVERSITÁRIO..... | 31 |
| 3.2. FASE 2 – COLETA AMOSTRAL DE ROTAS | 31 |
| 3.3. FASE 3 – ESTRUTURAÇÃO DE MODELOS: DEFININDO O MODELO DE PROGRAMAÇÃO LINEAR ADEQUADO..... | 32 |
| 3.4. FASE 4 – PROGRAMAÇÃO LINEAR: UTILIZAÇÃO DO EXCEL: SOLUÇÃO PELO SOLVER | 33 |
| 3.5. FASE 5 – PYTHON: UTILIZAÇÃO DA LINGUAGEM PARA UMA SOLUÇÃO MAIS ADEQUADA | 33 |
| 3.6. FASE 6 – CONFRONTO DAS SOLUÇÕES: COMPARAÇÃO DOS TRAJETOS OBTIDOS PELAS SOLUÇÕES E SELEÇÃO DO MAIS ADEQUADO | 34 |
| 4. RESULTADOS E DISCUSSÕES | 35 |
| 4.1. O PROBLEMA: SITUAÇÃO ATUAL DA ROTA DE TRANSPORTE UNIVERSITÁRIO..... | 35 |
| 4.2. COLETA AMOSTAL DE ROTAS..... | 36 |
| 4.3. PROGRAMAÇÃO LINEAR: UTILIZANDO O SOLVER PELO EXCEL | 40 |
| 4.4. CONFRONTO DAS SOLUÇÕES OBTIDAS: COMPARAÇÃO DOS TRAJETOS OBTIDOS PELAS SOLUÇÕES E SELEÇÃO DO MAIS ADEQUADO | 51 |
| 5. CONSIDERAÇÕES FINAIS..... | 54 |
| 6. REFERÊNCIAS BIBLIOGRÁFICAS | 56 |

1. INTRODUÇÃO

A Prefeitura Municipal de Timbaúba (PMT) entende que nem todos seus munícipes possuem as mesmas oportunidades, isto inclui a possibilidade de cursar e concluir o ensino superior. Para entregar mais acessibilidade a esses cidadãos universitários a PMT disponibiliza transportes (ônibus e vans) universitários que partem do centro da cidade sempre às 16hrs rumo às capitais da Paraíba e Pernambuco. Esses transportes realizam um determinado percurso todos os dias, orientados pela Secretaria de Educação de Timbaúba (SET), para entregar os alunos nas suas respectivas instituições de ensino e em tempo hábil para atingirem o primeiro horário de aulas do período noturno. A SET entende que os percursos determinados são as melhores alternativas para redução de tempo e, conseqüentemente, de custos, apenas pela experiência empírica relatada pelos motoristas.

Para a cidade de João Pessoa-PB são disponibilizados dois transportes universitários de segunda-feira à sexta-feira com partida da origem (Timbaúba-PE) no horário já informado anteriormente, sendo um desses transportes o responsável por entregar, em média diária, quarenta e cinco alunos de cinco instituições diferentes da rede pública e particular. Esse trabalho tem o foco justamente nesse transporte para cinco diferentes destinos em João Pessoa-PB, visando entregar uma solução matemática e de engenharia para a problemática de percurso que envolve pesquisa operacional, linguagem de programação e logística.

Para tal, irá utilizar-se de modelos de programação linear para o PCV de acordo com alguns conceitos e referências que serão mais aprofundadas ao decorrer do trabalho como Dantzig, Fulkerson e Johnson (1954) e Miller, Tucker e Zemlin (1960) além de entender uma visão sob a perspectiva de outros estudiosos do tema como Loesch e Hein (2008), Hillier e Lieberman (2006 e 2013), Arenales et al. (2007) e mais, para homologar definitivamente uma rota visando minimizar o tempo de traslado entre o ponto de partida e os pontos de interesse, conhecidos na teoria dos grafos como os arcos e nós.

Conta-se com o auxílio da ferramenta Excel e seu suplemento solver, ambos disponibilizados pela empresa Microsoft, para dar base na elaboração do problema matemático com modelo DFJ para o TSP, embora não seja a ferramenta viável para um problema com centenas de interações, essa solução será a primeira para definição de qual rodovia será escolhida para chegada em João Pessoa-PB.

Para abrilhantar o documento, entrega-se uma pequena demonstração da ótima rota em linguagem Python utilizando primeiramente a heurística do vizinho mais próximo e em seguida outro algoritmo, também em Python, construído com base no modelo MTZ (1960), onde este sim, tem uma maior complexidade e confiabilidade na entrega do percurso ideal.

Por último, serão comparadas as três soluções para entender qual a viabilidade do percurso atual e entregar a rota ideal para atendendo as restrições do problema e a função objetivo determinada.

1.1. JUSTIFICATIVA

O trabalho abordará o transporte universitário que segue a rota para João Pessoa-PB, podendo escolher, em primeira ordem, se segue pela BR230 ou pela BR101 para, em segundo momento, atingir as cinco instituições de ensino em tempo hábil para que o aluno (município) esteja presente no início do primeiro horário de aulas da noite. Portanto, em resumo:

- Definir qual será a rodovia escolhida como percurso, em seguida,
- Validar a rota ótima de destino
- Entregar os alunos em tempo hábil para o primeiro horário de aula noturno
- Entregar uma ferramenta de planejamento de rotas para a Prefeitura Municipal de Timbaúba (PMT) que poderá utilizá-la em outras problemáticas do município, aliando a teoria à prática e ajudando no desenvolvimento desse município e de sua comunidade.

1.2. OBJETIVOS

1.2.1. OBJETIVO GERAL

Homologar definitivamente uma rota visando minimizar o tempo de traslado entre o ponto de partida e os pontos de interesse pelo transporte universitário do município de Timbaúba-PE com destino a João Pessoa-PB para cinco instituições de ensino superior em tempo hábil para que todos os munícipes/discentes estejam presentes em todo seu horário de aulas do período noturno.

1.2.2. OBJETIVOS ESPECÍFICOS

- Analisar a rota usual (atual) do transporte universitário de Timbaúba-PE à João Pessoa-PB;
- Implementar um modelo matemático de programação linear adequado a problemática.
- Desenvolver um modelo heurístico computacional;
- Implementar um modelo computacional mais robusto que entregue a rota ótima
- Confrontar as rotas encontradas pelos cálculos realizados para viabilizar uma rota ótima ao transporte dos universitários timbaubenses.

2. FUNDAMENTAÇÃO TEÓRICA

2.1. PESQUISA OPERACIONAL

Diversos autores atribuem seus conceitos, alguns mais generalistas e outros mais detalhistas. Entretanto, todos eles deixam bastante claro: É impossível conversar sobre Pesquisa Operacional (PO) sem conversar sobre tecnologia, otimização e desenvolvimento. Moreira (2018, p.1) traz informações bastante concisas em seu livro Pesquisa Operacional: Curso introdutório, uma delas é sobre o termo Pesquisa Operacional que vem do inglês britânico *Operational Research* e foi aparentemente cunhado em 1938, para descrever o uso de cientistas na análise de situações militares. Também de acordo com Hillier; Lieberman (2006, p. 1), o surgimento da PO está diretamente relacionada as atividades militares da Segunda Guerra Mundial para se relacionarem e ratearem de maneira mais assertiva “os escassos recursos para diversas operações militares e atividades internas a cada operação.” Logaray, em Introdução à

Pesquisa Operacional (2013, p.1) explica que “...foi a partir da década de 1950, na economia do pós-guerra, que a PO teve grande espectro de aplicação nas empresas”. Portanto, ganhou notoriedade e pôde ser objeto de mais estudos para ramificar seu potencial de solução e suas operações.

A PO tornou-se essencial para o cenário tecnológico, uma vez que através dela tem-se a possibilidade de resolver problemas complexos com multivariáveis e dados. Arenales (2011, p.3) compartilha conosco sua percepção sobre a definição da PO.

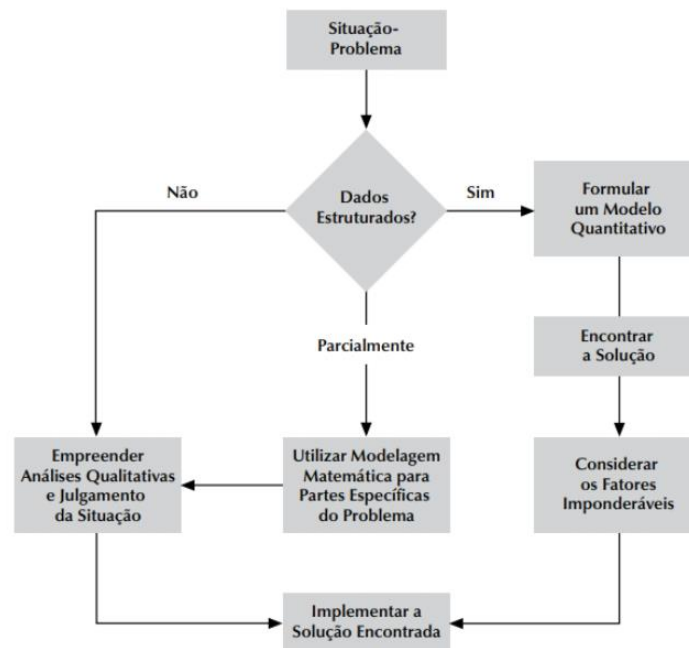
Consiste no desenvolvimento de métodos científicos de sistemas complexos, com finalidade de prever e comparar estratégias ou decisões alternativas. O objetivo é dar suporte à definição de políticas e determinação de ações de forma científica.

É notório como essa ferramenta auxilia as empresas e indústrias na melhor tomada de decisão, podendo prever cenários e destinar os recursos de maneira mais assertiva. Arenales (2011, p.3) ainda completa:

O componente tecnológico está relacionado a ferramentas de software e hardware para coletar e comunicar dados, e organizar esses dados, usando-os para gerar e otimizar modelos e reportar resultados, ou seja, a PO está se tornando um importante elemento nas metodologias de tecnologia da informação.

Sobre a natureza da PO, Hillier; Lieberman (2006, p. 2) afirmam que o processo começa na fase de observação do cenário, ou problema em questão, com o levantamento dos dados que se julgam necessários e relevantes. A próxima etapa é a construção de um modelo matemático e científico que tenta se assimilar ao máximo ao problema real. Entende-se então que este modelo representa suficientemente a realidade com as características precisas e essenciais da situação e que a conclusão dele terá total validade para sua aplicabilidade. Por conseguinte, é a fase de análise para promover os ajustes necessários e são realizados os testes da hipótese (validação do modelo). É basicamente o que mostra Longaray (2013, p.6), que informa: “Em sua forma mais simplificada, um modelo matemático é composto de variáveis, restrições, critérios e pelo menos um objetivo.”

FIGURA 1: Modulação de problemas para pesquisa operacional



Fonte: Moreira, 2004, p. 28 (adaptado).

A característica principal e o grande objetivo é que a PO procura sempre resolver o problema com uma melhor resposta possível, denominada solução ótima. E é justamente neste termo de “uma melhor resposta possível” e não “a melhor resposta possível” que se compreende o objetivo macro da PO, uma vez que, de toda forma há sempre uma melhoria a ser realizada, pois, os cenários se alteram e as soluções caminham em conjunto com eles.

Tratando-se de objetivos e modulação, é preciso entender o que significa as *variáveis de decisões* do problema.

Longaray (2013, p. 6) traz uma interessante visão predecessora do que seriam as variáveis de decisão e as classifica em variáveis controláveis e não controláveis, onde as variáveis controláveis são aquelas as quais o decisor poderá atuar para atingir seus objetivos e as não controláveis, intuitivamente será o oposto, quando o decisor não controla sua ação e ainda afetam diretamente nas consequências e nos resultados de uma tomada de decisão.

Como citada anteriormente, a *função objetivo*, é o fator decisório do problema, é o que irá indicar o que se deseja com determinada posição ou decisão.

Nos modelos formulados também são compostos de uma série de condições, as quais referem-se como *restrições*. Moreira (2018, p.16) as definem da seguinte maneira:

“As restrições dizem respeito à escassez de recursos, por um lado, e a limites impostos sobre nossas ações...”.

Essas restrições têm o direcionamento à maximização ou minimização da função objetivo do problema posto a discussão. São justamente as condições e peculiaridades que cada situação exige, são elas que devem ser respeitadas e ajustadas simetricamente com a realidade.

Retomando um pouco ao período pós segunda guerra, os muitos cientistas que haviam participado das equipes de PO motivaram-se para desenvolver mais sobre o assunto, resultando em avanços importantíssimos, um desses avanços é o reverenciado *método simplex* para solução de problemas com *programação linear* que observa-se um pouco a seguir.

2.2. PROGRAMAÇÃO LINEAR

Hillier; Lieberman (2013, p.42) afirmam que:

A programação linear usa um modelo matemático para descrever o problema em questão. O adjetivo linear significa que todas as funções matemáticas nesse modelo são necessariamente funções lineares. A palavra programação, nesse caso, não se refere à programação de computador; ela é, essencialmente, um sinônimo para planejamento. Portanto, a programação linear envolve o planejamento de atividades para obter um resultado ótimo, isto é, um resultado que atinja o melhor objetivo especificado (de acordo com o modelo matemático) entre todas as alternativas viáveis.

Portanto, entende-se que programação linear é um método matemático que consiste na solução de problemas com otimização. Essa otimização, como citada anteriormente, pode ser uma maximização ou minimização de uma situação problema.

No tópico anterior entende-se como se configura uma modulação de PO, e esta modulação é definitivamente uma programação linear onde informa-se o que se deseja obter, porém quais seriam as peculiaridades e condições para obtermos o resultado, ou seja, as restrições.

Entretanto a programação linear utiliza dos meios tecnológicos como os computadores e suas ferramentas de softwares e linguagens de programação. Para

completar o entendimento é preciso observar o relato de Moreira (2018, p.11) sobre programação linear:

A programação linear é, se não o mais popular, um dos modelos matemáticos mais populares, estruturado para resolver problemas que apresentem variáveis que possam ser medidas e cujos relacionamentos possam ser expressos por meio de equações e/ou inequações lineares.

Portanto, em resumo, tem-se uma formulação do problema para programação linear.

- Definir função objetivo – Maximizar ou minimizar determinada situação.
- Definir as restrições – Encontrar as condições e características do problema.
- Entender o modelo matemático – Respeitar as regras matemáticas sujeitas e defini-las também como restrições.

Loesch e Hein (2008) afirmam que todo problema de Programação Linear poderá ser definido de maneira genérica, seguindo:

Função objetivo

$$\{\text{Max, Min}\} Z = c_1x_1 + c_2x_2 + \dots + c_nx_n \quad (1)$$

Sujeito A

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \{=, \leq, \geq\} b_1 \quad (2)$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \{=, \leq, \geq\} b_2$$

...

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \{=, \leq, \geq\} b_m$$

Sendo

$$x_1, x_2, \dots, x_n \geq 0 \quad (3)$$

No modelo matemático mostrado, deve-se interpretar:

- x_1, x_2, \dots, x_n : É o conjunto de variáveis de decisões do problema;
- c_1, c_2, \dots, c_n : São os coeficientes da função objetivo;
- a_{ij} contidos em (2) são os coeficientes das restrições;

- b_j contidos em (2) são constantes chamadas de lado direito das

As restrições do modelo:

- Os sinais matemáticos $\{=, \leq, \geq\}$ significam a presença de uma dessas três relações em cada restrição proposta;
- A função objetivo expressa o objetivo geral do problema. Sendo esse objetivo para maximização ou de minimização;
- A expressão (3) diz respeito a restrição de não-negatividade das variáveis que constituem o problema.

Por isto, o problema é composto da seguinte maneira:

- (1) A função objetivo;
- (2) As restrições identificadas;
- (3) As restrições de não negatividade.

Longaray (2013, p.70) alerta que a programação linear se esgueira em quatro condições (hipóteses) que devem ser satisfeitas para, ao serem empregadas, alcançar o resultado almejado. Longaray resume essas hipóteses, de tal forma:

- Hipótese da proporcionalidade: *em modelos de programação linear, presume-se que a contribuição de cada atividade ao valor de Z é proporcional ao nível de atividade x_j , representado por $c_j x_j$ na função objetivo. Da mesma forma, a contribuição de cada atividade do lado esquerdo de cada restrição de recurso é proporcional ao nível de atividade x_j , representado pelo termo $a_{ij} x_j$ na restrição.*
- Hipótese da aditividade: *a condição de aditividade, existente na totalidade dos modelos de programação linear, consiste em considerar as atividades do modelo entidades absolutamente independentes, não permitindo que haja interdependência entre elas.*
- Hipótese da divisibilidade: *as variáveis de decisão de um modelo de programação linear podem assumir quaisquer valores, até mesmo valores não inteiros (fracionários), que atendam às restrições de recursos e de não negatividade.*
- Hipótese da certeza: *essa hipótese presume que o valor atribuído a cada parâmetro de um modelo de programação linear é assumido como uma constante conhecida. São parâmetros os coeficientes c_j na função objetivo, os coeficientes a_{ij} nas restrições de recursos, e os coeficientes b_i no lado direito das restrições de recursos.*

É importante salientar que a Programação Linear é comumente utilizada na alocação de recursos e serviços, entretanto sua aplicabilidade é muito vasta e se estende

basicamente a todos os cenários possíveis dentro de indústrias, corporações e organizações públicas.

2.3. MÉTODOS HEURÍSTICOS

Russell; Norvig (2013), em seu livro "Inteligência Artificial: Uma Abordagem Moderna", definem heurísticas como "regras de ouro", ou seja, regras práticas que podem ajudar a encontrar soluções em situações em que uma abordagem algorítmica completa seria impraticável. Eles destacam que os métodos heurísticos são frequentemente usados em problemas de busca e otimização, permitindo encontrar soluções razoáveis em tempo hábil. Já Simon (1969) um dos pioneiros da inteligência artificial e ganhador do Prêmio Nobel de Economia, enfatiza a importância das heurísticas na tomada de decisões humanas. Ele argumenta que, devido à complexidade do mundo real, os humanos frequentemente recorrem a heurísticas simples e eficientes para resolver problemas de forma satisfatória, mesmo que não seja a solução ótima.

Polyag (2004) em seu livro "How to Solve It", aborda heurísticas como estratégias mentais para resolver problemas matemáticos. Ele destaca a importância de técnicas heurísticas, como a decomposição do problema em partes menores, a busca por padrões e a experimentação, para encontrar soluções. Ainda argumenta que as heurísticas são essenciais para estimular a criatividade e a resolução de problemas.

Melanie Mitchell (2009), em seu livro "Complexity: A Guided Tour", explora o papel das heurísticas na resolução de problemas em sistemas complexos. Ela destaca que:

... os métodos heurísticos são amplamente utilizados em algoritmos de inteligência artificial, como algoritmos genéticos e redes neurais, para encontrar soluções aproximadas em espaços de busca vastos e complexos.

Kahneman; Tversky (1974), em suas pesquisas sobre heurísticas e vieses cognitivos, identificam várias heurísticas que as pessoas usam na tomada de decisões, como a heurística da disponibilidade e a heurística da representatividade. Eles mostram como essas heurísticas podem levar a julgamentos e decisões subótimos, destacando a interação entre racionalidade limitada e heurísticas na mente humana.

Glover (2003), em Handbook of Metaheuristics, categoriza os métodos heurísticos em diversas classes, incluindo:

- Busca Local: Métodos que exploram uma região do espaço de solução em busca de soluções melhores, fazendo pequenas alterações em soluções existentes.
- Algoritmos Genéticos: Inspirados no processo de evolução natural, esses métodos exploram soluções candidatas através de operadores genéticos, como mutação, recombinação e seleção.
- Algoritmos de Enxame: Modelam o comportamento de sistemas coletivos, como enxames de pássaros ou colônias de formigas, para explorar o espaço de busca.
- Otimização por Colônia de Formigas: Baseada no comportamento das formigas na busca de alimento, essa abordagem utiliza feromônios para guiar a busca por soluções melhores.
- Otimização por Enxame de Partículas: Modela o comportamento de um enxame de partículas em busca de soluções ótimas, onde cada partícula representa uma solução candidata.

Esses autores oferecem diferentes perspectivas sobre métodos heurísticos, destacando sua importância em diversos campos do conhecimento, desde inteligência artificial até psicologia cognitiva. Suas obras são referências importantes para compreender a natureza e o papel das heurísticas na resolução de problemas.

2.4. CAIXEIRO VIAJANTE

O PCV consiste em uma situação em que existe um conjunto de cidades, um ponto de partida e um caixeiro. Esse caixeiro sai desse ponto de partida (sede) visita todas as cidades ou um subconjunto de cidades, somente uma única vez, e retorna ao ponto de partida visando otimizar seu percurso e tempo de execução.

O Problema do Caixeiro Viajante (PCV) é um problema clássico de otimização combinatória. Arenales et. al (2007) em seu livro “Pesquisa Operacional”, ensina que o Problema do Caixeiro Viajante (PCV) é um problema que tenta determinar a menor rota para percorrer uma série de cidades (visitando uma única vez cada uma delas),

retornando à cidade de origem. Ele é um problema de otimização NP-difícil inspirado na necessidade dos vendedores em realizar entregas em diversos locais (as cidades) percorrendo o menor caminho possível, reduzindo o tempo necessário para a viagem e os possíveis custos com transporte e combustível.

Ainda segundo Arenales et. al. (2007) problemas de caixeiro viajante pertencem à classe de problemas de roteamento de nós, e são definidos em grafos orientados ou não orientados.

Métodos heurísticos são frequentemente aplicados para encontrar soluções aproximadas para o PCV, uma vez que encontrar a solução exata é impraticável em muitos casos devido à sua complexidade. Tal complexidade é provada quando se obtém numa solução o que denomina-se de sub-rotas. Refere-se como sub-rotas no contexto do PCV, os caminhos parciais que o caixeiro viajante faz durante sua jornada. O caixeiro viajante deve visitar todas as cidades exatamente uma vez, mas ele pode fazer isso em qualquer ordem. Portanto, o processo de encontrar a rota mais curta envolve considerar diferentes combinações de cidades que formam sub-rotas. Essas sub-rotas são exploradas durante o processo de busca para encontrar a solução ótima para o problema. Existem várias abordagens algorítmicas para resolver o PCV, como a busca exaustiva, programação dinâmica, algoritmos genéticos, entre outros. Cada uma dessas abordagens lida de maneira diferente com as sub-rotas, mas todas compartilham o objetivo comum de encontrar a rota mais curta que visita todas as cidades.

Diversas propostas de formulações matemáticas para eliminação de sub-rotas, têm sido desenvolvidas desde muito tempo podendo destacar Dantzig, Fulkerson e Johnson (DFJ, 1954) e para Miller, Tucker e Zemlin (MTZ, 1960), que são consideradas clássicas para o Problema do Caixeiro Viajante. Diversos problemas de decisão de ordem prática podem ser classificados como problemas de fluxo de rede. Esses têm uma característica em comum – podem ser descritos ou representados por um grafo (Ragsdale, 2018).

De acordo com Miller, Tucker e Zemlin (1960), o problema do Caixeiro Viajante segue a seguinte formulação matemática:

Dados:

C_{ij} : Distância de i até j

N : Instância

i : Origem

j: Destino

X: Local

FIGURA 2: FORMULAÇÃO MATEMÁTICA DO CAIXEIRO VIAJANTE

$$\begin{aligned} \min \quad & \sum_{i=1}^n \sum_{j=1, j \neq i}^n c_{ij} x_{ij}, \\ \text{sujeito a} \quad & \sum_{i=1, i \neq j}^n x_{ij} = 1, \quad j = 1, 2, \dots, n, \\ & \sum_{j=1, j \neq i}^n x_{ij} = 1, \quad i = 1, 2, \dots, n, \\ & u_i - u_j + nx_{ij} \leq n - 1, \quad 2 \leq i \neq j \leq n, \\ & x_{ij} \in \{0, 1\} \quad i, j = 1, 2, \dots, n, \quad i \neq j, \\ & u_i \in \mathbb{R}^+ \quad i = 1, 2, \dots, n. \end{aligned}$$

Fonte: Miller, Tucker e Zemlin (1960)

Na modulação DFJ (1954) observa-se uma pequena diferenciação na modelagem das restrições quando se trata na restrição de eliminação de subrotas, que segue:

Variáveis de decisão:

C_{ij}: Distância de i até j

N: Instância

i: Origem

j: Destino

$$x_{ij} = \begin{cases} 1, & \text{se visita } j \in N \text{ após } i \in N; \\ 0, & \text{caso contrário;} \end{cases}$$

Função Objetivo: Minimizar / Maximizar

$$\sum_{i=1}^n \sum_{j=1}^n C_{ij} X_{ij} \quad (1)$$

Restrições:

$$\sum_{i=1, i \neq j}^n x_{ij} = 1, \quad \forall j \in N \quad (2)$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1, \quad \forall i \in N \quad (3)$$

$$x_{ij} \in \{0,1\}, \quad \forall ij \in N \quad (4)$$

É preciso notar que somente as restrições de “entrada” e de “saída” não garantem a formação de um ciclo hamiltoniano, ou seja, um ciclo sequencial partindo da origem passando local a local até o último ponto (destino). Para isto, utiliza-se a restrição para eliminação de sub-rotas ou subcaminhos de Dantzig, Fulkerson e Johnson et.al. (1954)

$$\sum_{(i,j) \in S} x_{ij} \leq |S| - 1, \quad \forall S \subset V, S \neq \emptyset \quad (5)$$

2.5. ALGUNS CONCEITOS DE LOGÍSTICA

Chapman et al. (1992) em *The Fundamentals of Logistics*, logística é o processo de gerenciar o fluxo de bens, informações e serviços, desde a fonte de produção até o

ponto de consumo, de maneira eficiente e eficaz. Isso envolve atividades como planejamento, implementação e controle de transporte, armazenamento, distribuição, gerenciamento de inventário, processamento de pedidos e informações, com o objetivo de atender às necessidades dos clientes enquanto minimiza custos e maximiza o valor.

Já, Dias (2016), em seu livro *Introdução à Logística - Fundamentos, Práticas e Integração*, explica que

A logística administra os recursos de toda movimentação de recursos materiais e equipamentos da empresa, coordenando a compra, a movimentação, a armazenagem, o transporte e a distribuição física, assim como gerenciando todas as informações de cada fase do processo.

Para Bowersox et al. (2007) logística é a responsabilidade de projetar e administrar sistemas para controlar o transporte e a localização geográfica dos estoques de materiais e produtos inacabados e acabados pelo menor custo total.

Através destes conceitos pode-se inferir que o objetivo primário da logística é garantir que os produtos certos estejam disponíveis no local certo, no momento certo e na quantidade certa, ao menor custo possível. Isso envolve a coordenação de diversas operações ao longo de toda a cadeia de suprimentos, desde a obtenção de matérias-primas até a entrega do produto ao cliente.

Ainda Dias (2016), entrega uma percepção sobre a avaliação dos meios de transportes que estão presentes diariamente em nossa realidade e a escolha necessária para cada situação em específico. Ele informa que é preciso analisar vantagens x desvantagens e mostra que os indicadores mais formais surgem a partir do confronto das informações entre distância, custo, tempo e segurança.

Este trabalho irá se debruçar um pouco mais sobre o conceito de transporte logístico com o modal rodoviário, que será fundamental para o entendimento do que se trata o problema do caixeiro viajante e toda questão envolva sobre transporte de pessoas buscando um caminho mais curto.

2.5.1 MODAL RODOVIÁRIO

O modal rodoviário é uma das modalidades de transporte de carga e passageiros que utiliza vias terrestres, como estradas e rodovias, para movimentar mercadorias, pessoas e veículos. É um dos modos de transporte mais flexíveis e amplamente

utilizados em muitas partes do mundo, devido à sua acessibilidade e capacidade de alcançar áreas remotas.

De acordo com o autor Razzolini Filho (2007) o modal de transporte mais utilizado no Brasil é o modal rodoviário. Para se ter uma ideia da diferença entre esse modal e os demais, basta dizer que a malha rodoviária instalada no País (estradas) representa cerca de 20% do total do território.

Com este adendo é necessário evidenciar as vantagens e desvantagens deste modal. Sobre esse assunto, Caxito (2019) traz à tona da seguinte maneira:

“As vantagens:

- Adequado para curtas e médias distâncias.
- Simplicidade no atendimento das demandas e agilidade no acesso às cargas.
- Menor manuseio da carga e menor exigência de embalagem.
- O desembarço na alfândega pode ser feito pela própria empresa transportadora.
Atua de forma complementar aos outros modais, possibilitando a intermodalidade e a multimodalidade.
Permite as vendas do tipo entrega porta a porta, trazendo maior comodidade para exportador e importador.”

As desvantagens:

- Fretes mais altos em alguns casos.
- Menor capacidade de carga entre todos os outros modais.
- Menos competitivo para longas distâncias”.

Este modal é de suma importância para locomoção de bens e realizações de serviços no mundo inteiro. O transporte de estudantes pelo modal rodoviário é uma prática comum em todo o mundo, especialmente em áreas onde o acesso às instituições educacionais requer deslocamento de longa distância.

É um serviço fornecido principalmente pelos governos municipais e estaduais, essencial na contribuição do acesso à educação da sociedade como um todo.

2.6 PROGRAMAÇÃO: PEQUENA ABORDAGEM SOBRE A LINGUAGEM PYTHON

Python é uma linguagem de programação de alto nível, interpretada, de propósito geral e de código aberto.

Guido van Rossum começou a trabalhar no desenvolvimento do Python no final dos anos 1980, enquanto estava no Centrum Wiskunde & Informatica (CWI) nos Países Baixos. Ele queria criar uma linguagem de programação fácil de ler e entender, influenciada pela linguagem ABC. A primeira versão do Python, a 0.9.0, foi lançada em fevereiro de 1991.

Programação em Python refere-se ao processo de escrever código em Python, uma linguagem de programação de alto nível conhecida por sua simplicidade e legibilidade. Python é uma linguagem versátil amplamente utilizada em uma variedade de domínios, incluindo desenvolvimento web, ciência de dados, automação, inteligência artificial, aprendizado de máquina e muito mais. Sua sintaxe limpa e fácil de entender torna Python uma escolha popular para iniciantes e profissionais experientes.

Sweigart (2015, p.34), em *Automate the Boring Stuff with Python*, explica:

“Python se refere à linguagem de programação (com regras de sintaxe para escrever o que é considerado um código Python válido) e ao software do interpretador Python, que lê o código-fonte (escrito na linguagem Python) e executa suas instruções.

O nome Python é proveniente do grupo surreal de comédia britânico Monty Python”.

A versão 1.0 do Python foi lançada em janeiro de 1994, marcando um ponto significativo em seu desenvolvimento. Esta versão inicial já possuía muitas das características fundamentais que tornam Python popular hoje, como exceções, funções lambda e manipulação de exceções.

Nos anos seguintes, Python começou a ganhar popularidade gradualmente entre os desenvolvedores. Sua sintaxe limpa, legibilidade e versatilidade atraíram muitos programadores. A simplicidade da linguagem e sua ampla gama de bibliotecas e frameworks contribuíram para sua crescente adoção em uma variedade de domínios.

Em 2000, o Python 2.0 foi lançado, introduzindo recursos como a funcionalidade de geradores e a declaração 'with'. Posteriormente, em 2008, o Python 3.0 foi lançado com mudanças significativas na linguagem, focadas na resolução de inconsistências e melhorias de design. No entanto, a transição do Python 2 para o Python 3 foi gradual devido a algumas incompatibilidades entre as versões.

Python construiu uma comunidade vibrante ao longo dos anos, com contribuidores de todo o mundo. Isso resultou em um vasto ecossistema de bibliotecas, frameworks e ferramentas que impulsionam o desenvolvimento de software em várias áreas, incluindo desenvolvimento web, ciência de dados, automação, inteligência artificial e muito mais.

Atualmente, Python é uma das linguagens de programação mais populares do mundo. É amplamente utilizado em empresas de tecnologia, instituições acadêmicas, startups e em muitos outros lugares. Sua simplicidade, legibilidade e flexibilidade continuam atraindo novos usuários e contribuidores para a comunidade Python.

3 METODOLOGIA

Neste capítulo é apresentado os procedimentos metodológicos realizados durante o trabalho, sobre a introdução das informações do tipo de pesquisa, planejamento, a coleta de dados, o manejo e tratamento dos dados e as etapas para alcançar tais indicativos dispostos nos resultados.

Este estudo é de natureza aplicada, uma vez que busca encontrar uma solução para minimizar o tempo de um determinado traslado utilizando um modelo de pesquisa operacional que será indicado para realização na prática pela Secretaria de Transportes do Município.

Os objetivos estão definidos em exploratórios e normativos, pois a intenção do trabalho é estudar e viabilizar, definir ou otimizar uma determinada rota de transporte universitário buscando padronizar o modelo para alcançar a redução do lead time de viagem. A abordagem é quantitativa, porque todas as informações trazidas neste documento serão indicativos matemáticos, ou seja, em números. E quanto aos procedimentos determinados serão pesquisas documentais e o estudo de caso avaliativo. O estudo de caso é uma estratégia de pesquisa baseada em evidências, e tem o intuito de verificar a relação entre conteúdos e testar teorias baseada nos casos (Eisenhardt,1989). Avaliativo pois, o objetivo principal é utilizar os dados para avaliar o mérito da prática em questão.

A seguir, estão descritas as etapas para o desenvolvimento do trabalho:

- Fase 1 - Situação atual da rota do transporte universitário.
- Fase 2 - Coleta amostral de rotas.
- Fase 3 - Definindo o modelo de programação linear adequado.
- Fase 4 - Utilização do Excel: Solução pelo solver.
- Fase 5 - Python: Utilização da linguagem para uma solução mais adequada.

- Fase 6 - Confronto das soluções: Comparação dos trajetos obtidos pelas soluções e seleção do mais adequado

3.1. FASE 1 – O PROBLEMA: SITUAÇÃO ATUAL DA ROTA DO TRANSPORTE UNIVERSITÁRIO

O problema se trata de um roteamento de veículo que busca a melhor rota para o transporte de estudantes universitários da cidade de Timbaúba-PE no período noturno com destino à cinco instituições de ensino diferentes em João Pessoa-PB. Os alunos devem estar presentes nas suas respectivas universidades (locais) em tempo hábil para gozarem do primeiro horário de aulas. O transporte tratado neste trabalho é um ônibus escolar de marca Volkswagen e modelo ORE 3 (Ônibus Rural Escolar 3) comportando transportar até 59 (cinquenta e nove) estudantes sentados, dados de acordo com a FNDE (2013).

Na primeira fase verificou-se como o município opera a demanda e como é realizado o transporte dos alunos aos locais de destino, bem como a infraestrutura das rodovias, quais são os caminhos possíveis, também de variabilidades que podem atrasar a viagem, como as questões de trânsito ou do próprio transporte. Essas observações foram realizadas pelo autor deste trabalho em conjunto com o motorista do ORE 3.

Nesta fase é que houve contribuição dos passageiros e dos motoristas para o desenvolvimento do documento. O esforço de observação, empírico, atrelados a experiência do discente, autor deste documento, e auxílio de colegas conterrâneos são fundamentais para a obtenção de dados consistentes relacionados a elaboração do caso. A SET não informou todos os dados e os que disponibilizou não possuem total exatidão.

3.2. FASE 2 – COLETA AMOSTRAL DE ROTAS

Nesta fase contou-se com auxílio dos próprios usuários (passageiros) do ônibus universitário e do motorista que possui autoridade e experiência para tratar do assunto.

A SET também determina uma rota que julga, sem estudo ou dados comprovados, ser a ótima, que é percorrida já há mais de dois anos ininterruptos, com alteração dela apenas em casos de imprevisibilidades que podem ocorrer com o veículo, rodovia ou trânsito.

Em dois dias distintos percorreu-se duas rotas diferentes pelas rodovias possíveis para identificar o caminho mais curto, por conseguinte realizada a cronometragem dos tempos de um local a outro por dois colegas, um aluno da UNIPE e outro aluno da FPB, ambos marcaram e anotaram o tempo de percurso de um ponto a outro iniciando o cronometro na partida e zerando-o assim que chegava no próximo local. Outra cronometragem, de tempo total de percurso, foi utilizada, iniciando a contagem em Timbaúba-PE e parando na FPB. Da mesma forma foi realizado no retorno de João Pessoa-PB à Timbaúba-PE.

A partir destas contagens foram montadas as matrizes distâncias que seguem no tópico 4. Resultados.

3.3. FASE 3 – ESTRUTURAÇÃO DE MODELOS: DEFININDO O MODELO DE PROGRAMAÇÃO LINEAR ADEQUADO.

A escolha do modelo de programação, DFJ (1954) para modulação no solver e MTZ para linguagem robusta em Python, além da heurística do vizinho mais próximo foram realizadas com base nas exigências identificadas durante a análise do percurso, já que os métodos de Programação Linear se ajustam às condições de acordo com as diversas variáveis do problema e suas interações.

É nessa fase que é essencial aplicar os conhecimentos teóricos adquiridos em pesquisa operacional para desenvolver um modelo que faça uso de dados quantitativos visando a geração de rotas ótimas e se encaixe com o propósito da situação.

Nesta fase, também é crucial examinar as opções de decisão disponíveis, identificar as limitações, avaliar as demandas do sistema e estabelecer as restrições. No caso da proposta formulada, é necessário determinar se o modelo é pré-existente ou se requer relações matemáticas mais complexas.

O PCV entrega uma rota fechada, ou seja, ciclo hamiltoniano, como já explicado no tópico 2.4, entretanto nesta situação em particular será quebrado um pouco desse paradigma. A partir do momento em que encontra-se um ciclo (ou rota) na ida será considerado o percurso total sendo da origem (0-Timbaúba-PE) até a última instituição que o ônibus entrega os alunos e aguarda o final do período de aulas para, a partir dali, então considerando neste momento a última instituição como origem, realizar o caminho

de retorno para buscar os alunos nas suas faculdades e partirem de João Pessoa-PB com direção à Timbaúba-PE pontualmente às 22hrs com uma margem de erro de cinco minutos mais cedo ou mais tarde.

3.4. FASE 4 – PROGRAMAÇÃO LINEAR: UTILIZAÇÃO DO EXCEL: SOLUÇÃO PELO SOLVER

A primeira solução proposta para o problema, de maneira mais simples e manual, uma vez que é necessário expressar na ferramenta, matematicamente, as restrições e função objetivo do problema.

O solver irá calcular as primeiras rotas possíveis. Exibem-se, neste momento, quatro matrizes as quais se tratam de dados cronometrados e tempo de percurso calculado pelo google maps, para cada rodovia (BR230 e BR101), por conseguinte, serão encontradas quatro soluções ótimas, sendo a matriz que mais atender a função objetivo de minimização do tempo de percurso será a selecionada para o cálculo nas demais métricas do trabalho em Python.

Esses resultados irão trazer as primeiras impressões e informações, onde será selecionada a que mais atende aos parâmetros requisitados e utilizada como base para a fase seguinte e conseqüentemente conclusão do trabalho.

3.5. FASE 5 – PYTHON: UTILIZAÇÃO DA LINGUAGEM PARA UMA SOLUÇÃO MAIS ADEQUADA

Utilizando a linguagem Python atrelada aos conceitos teóricos de pesquisa operacional serão criados algoritmos para entender como se comporta o problema, de forma, também, a inferir sobre a solução caso seja a mesma da fase anterior ou altere-se.

A partir do resultado calculado em solver pelo método simplex, a matriz selecionada será pilar da estrutura algorítmica na heurística do vizinho mais próximo e da implementação, também em Python, de um algoritmo pautado no método MTZ (1960).

Nesta fase é exibido a real intenção do trabalho, incorporando um método e modelo mais robusto, conseqüentemente mais confiável e viável para o PCV que é NP-difícil, e resolver a problemática para então comprovar a melhor rota e outorga-la.

3.6. FASE 6 – CONFRONTO DAS SOLUÇÕES: COMPARAÇÃO DOS TRAJETOS OBTIDOS PELAS SOLUÇÕES E SELEÇÃO DO MAIS ADEQUADO

O propósito desta etapa é contrastar as rotas derivadas das amostras com as simulações realizadas no solver pelo simplex e em Python, a fim de determinar se as soluções são iguais ou não.

4. RESULTADOS E DISCUSSÕES

4.1. O PROBLEMA: SITUAÇÃO ATUAL DA ROTA DE TRANSPORTE UNIVERSITÁRIO

Como já foi bastante discutido em tópicos anteriores pretende-se entender se a rota que o transporte universitário do município de Timbaúba-PE percorre até João Pessoa-PB é a ideal para entregar seus munícipes, alunos, nas devidas instituições reduzindo ao máximo o tempo de percurso e conseqüentemente o tempo de entrega desses universitários comparando três métodos de solução para compreender qual o mais completo e determiná-lo.

As instituições que possuem alunos matriculados e estão na rota do transporte são as seguintes:

- Centro Universitário de João Pessoa (UNIPÊ)
- Universidade Federal da Paraíba (UFPB)
- Faculdade da Paraíba (FPB)
- Centro Universitário Maurício de Nassau (UNINASSAU)
- Faculdade Três Marias (FTS).

Deve-se analisar primeiramente dois caminhos distintos com destino à João Pessoa-PB e retorno à Timbaúba-PE definidos previamente pela Secretaria de Educação de Timbaúba-PE:

1- Rota 1: Passando pelo município de Ferreiros-PE, Camutanga-PE e Juripiranga pela PE082 na BR408, seguindo pela PE075 passando por Santa Rita-PB e Bayeux-PB pela BR230 até as universidades e retornando à origem pela mesma rota 1.

2- Rota 2: Passando pelo município de Ferreiros-PE, Camutanga-PE e Juripiranga pela PE082 na BR408, seguindo pela PE075 até os municípios de Itambé-PE, entrando na PB032 no município de Pedras de Fogo-PB atingindo a BR101 em Alhandra-PB e por fim até as universidades e retornando à origem pela mesma rota 2.

4.2. COLETA AMOSTAL DE ROTAS

Realizada de forma autônoma, buscando informações com órgãos competentes como a SET e a PMT, entretanto não foram compartilhados dados mais precisos.

Obteve-se informações do quantitativo geral de alunos (informação não sigilosa). Sobre a quantidade de transportes disponíveis, horários, alunos de cada instituição, e cronometragem do tempo de viagem de cada ponto a outro foi obtido de forma empírica, observação pela experiência do concluinte que utiliza dos transportes desde fevereiro de 2014 e auxílio do Google Maps.

Foi observado durante dez dias úteis que o ônibus universitário partia do destino, Praça do Centenário no município de Timbaúba-PE, em todos os dias às 16hrs com o total de 45 alunos (média diária) com destino às cinco universidades de João Pessoa-PB.

Com relação ao tempo de viagem, foi cronometrado em dois dias úteis de maior demanda de alunos na segunda-feira e terça-feira (18 e 19 de março de 2024), com auxílio de um colega do município que utiliza o transporte, discente da UFPB, que se propôs a marcar o tempo da origem até os destinos e o motorista do ônibus que realizou duas rotas de viagens, uma pela BR101 na segunda e outra pela BR230 na terça. Importante salientar que além do percurso anteriormente citado, foi preciso realizar deslocamentos em carro particular saindo da origem às 16hrs como demanda a SET e percorrendo nos caminhos com mesmo fluxo no período em que transita o ônibus universitário diariamente na data 22/03/2024 (vinte e dois de março de dois mil e vinte e quatro), para completar as matrizes de cronometragem e confirmar a veracidade dos tempos cronometrados pelo universitário nos dias acima mencionados.

Foram utilizados duas cronometragens simultâneas:

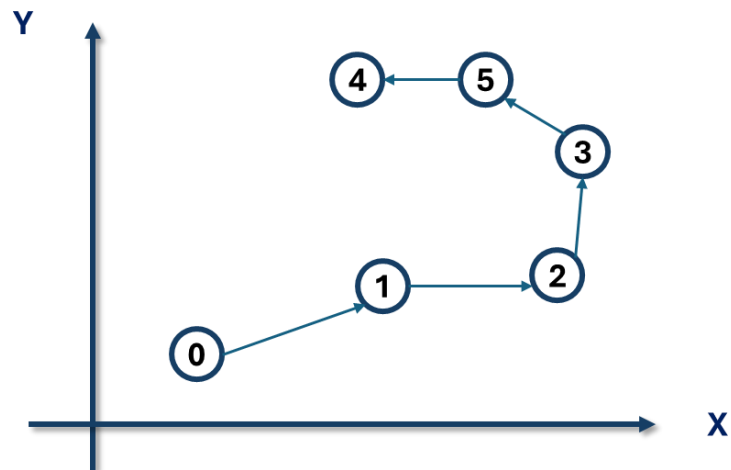
1- Tempo total de viagem desde a origem: Onde foi iniciado o cronômetro no momento da partida da origem até o último destino (instituição) o tempo de viagem.

2- Origem e destino: Início da contagem na partida da origem e parada no primeiro destino, início de contagem do primeiro destino e parada no segundo e assim sucessivamente até o último destino, por conseguinte no retorno dessa rota, invertendo no segundo dia (terça-feira) os caminhos.

Atualmente a rota de ida mais utilizada pelo transporte universitário é a rota [0-1-2-3-5-4] com tempo de percurso que varia entre 110 minutos pela BR230 e 139 minutos

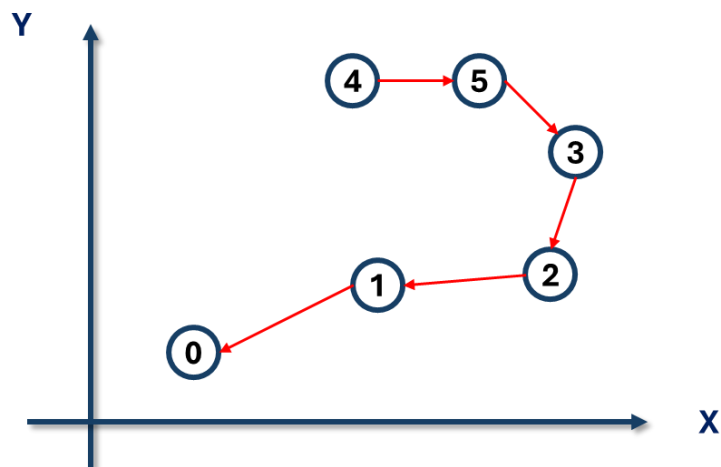
pela BR101. O transporte aguarda no ponto 4 (FPB) até as 21:50hrs para realizar o trajeto de retorno à origem inicial (Timbaúba-PE), com previsão de chegada às 00hrs em Timbaúba-PE, representada pela rota [4-5-3-2-1-0] com tempo total de percurso com 106 minutos. Essas rotas estão representadas pelos grafos abaixo:

FIGURA 3: Grafo da rota de ida comum realizada (Timbaúba-UNIPE-UFPB-Uninassau-FTM-FPB)



Fonte: Autor.

FIGURA 4: Grafo da rota de retorno comum realizada (FBP-FTM-Uninassau-UFPB-UNIPE-Timbaúba)



Fonte: Autor.

Agora os dados coletados serão analisados tomando como referência a rota comum, demonstrada nos grafos acima. Serão postos em três cálculos diferentes para validação da mesma ou otimização de acordo com os resultados obtidos pelos métodos expostos.

Com auxílio do Google Maps observa-se a média de tempo de instituição para instituição e confrontar com o cronometrado. Segue as tabelas abaixo dos resultados de cronometragem de segunda e terça-feira, por conseguinte as tabelas construídas com auxílio do google maps, todas com unidade de medida em minutos.

TABELA 1 – CRONOMETRAGEM (BR101)

| | TIMBAÚBA | UNIFE | UFPB | UNINASSAU | FPB | TRES MARIAS |
|--------------------|-----------------|--------------|-------------|------------------|------------|--------------------|
| TIMBAÚBA | 0 | 100 | 115 | 126 | 122 | 119 |
| UNIFE | 90 | 0 | 9 | 18 | 21 | 26 |
| UFPB | 98 | 6 | 0 | 11 | 15 | 13 |
| UNINASSAU | 112 | 16 | 10 | 0 | 19 | 9 |
| FPB | 107 | 19 | 16 | 13 | 0 | 2 |
| TRES MARIAS | 109 | 24 | 12 | 7 | 3 | 0 |

Fonte: Autor

TABELA 2 – CRONOMETRAGEM (BR230)

| | TIMBAÚBA | UNIFE | UFPB | UNINASSAU | FPB | TRES MARIAS |
|--------------------|-----------------|--------------|-------------|------------------|------------|--------------------|
| TIMBAÚBA | 0 | 85 | 93 | 104 | 91 | 95 |
| UNIFE | 81 | 0 | 8 | 19 | 23 | 24 |
| UFPB | 88 | 6 | 0 | 11 | 14 | 13 |
| UNINASSAU | 99 | 17 | 9 | 0 | 17 | 8 |
| FPB | 84 | 20 | 15 | 12 | 0 | 3 |
| TRES MARIAS | 87 | 23 | 11 | 7 | 4 | 0 |

Fonte: Autor

TABELA 3 – TEMPO DE VIAGEM PELO GOOGLE MAPS (BR101)

| | TIMBAÚBA | UNIPE | UFPB | UNINASSAU | FPB | TRES MARIAS |
|--------------------|-----------------|--------------|-------------|------------------|------------|--------------------|
| TIMBAÚBA | 0 | 90 | 100 | 108 | 106 | 104 |
| UNIPE | 94 | 0 | 9 | 19 | 24 | 22 |
| UFPB | 95 | 7 | 0 | 9 | 11 | 10 |
| UNINASSAU | 107 | 15 | 9 | 0 | 5 | 7 |
| FPB | 104 | 18 | 14 | 10 | 0 | 6 |
| TRES MARIAS | 102 | 20 | 10 | 5 | 6 | 0 |

Fonte: Autor

TABELA 4 – TEMPO DE VIAGEM PELO GOOGLE MAPS (BR230)

| | TIMBAÚBA | UNIPE | UFPB | UNINASSAU | FPB | TRES MARIAS |
|--------------------|-----------------|--------------|-------------|------------------|------------|--------------------|
| TIMBAÚBA | 0 | 83 | 90 | 101 | 89 | 91 |
| UNIPE | 80 | 0 | 9 | 19 | 24 | 22 |
| UFPB | 88 | 7 | 0 | 9 | 11 | 10 |
| UNINASSAU | 98 | 15 | 9 | 0 | 5 | 7 |
| FPB | 86 | 18 | 14 | 10 | 0 | 6 |
| TRES MARIAS | 89 | 20 | 10 | 5 | 6 | 0 |

Fonte: Autor

A ferramenta Excel da empresa Microsoft com seu complemento solver será utilizada para entendermos melhor como funciona a ideia de solução do caixeiro viajante para dar base na composição da linguagem de programação em Python.

A partir do melhor aproveitamento de tempo, ou seja, da função objetivo que mais atende a minimização do tempo de viagem, selecionar a matriz que será a base para a solução global na linguagem Python.

O foco é confrontar as informações e soluções do Excel e dos dois modelos estruturados em Python para analisarmos os resultados e avaliar se aquele cujo método e função objetivo melhor atende às restrições do problema.

Em ambas as soluções foi enfrentado um problema com o surgimento de sub-caminhos ou sub-rotas.

As sub-rotas são consideradas durante o processo de busca de soluções para esse problema. Essas sub-rotas podem ser exploradas para determinar qual delas resulta na rota global mais curta. Isso geralmente é feito utilizando algoritmos de otimização, como o algoritmo de busca em profundidade, o algoritmo genético ou o algoritmo de programação dinâmica. Esses algoritmos examinam diferentes combinações de sub-rotas para determinar a rota global mais eficiente.

4.3. PROGRAMAÇÃO LINEAR: UTILIZANDO O SOLVER PELO EXCEL

Os locais estão configurados da seguinte forma, de acordo com a tabela abaixo:

TABELA 5: CONFIGURAÇÃO DOS LOCAIS

| | | | | | |
|----------|-------|------|-----------|-----|-------------|
| 0 | 1 | 2 | 3 | 4 | 5 |
| TIMBAÚBA | UNIFE | UFPB | UNINASSAU | FPB | TRÊS MARIAS |

Fonte: Autor

Através da solução pelo solver obtiveram-se as seguintes rotas e tempo de viagem, calculado pela função objetivo do problema:

- Tempo de percurso: 114 minutos

TABELA 6: ROTA BR230 COM REFERÊNCIA DO GOOGLE MAPS

| | | | | | |
|----------|-----|-------------|-----------|------|-------|
| 0 | 4 | 5 | 3 | 2 | 1 |
| TIMBAÚBA | FPB | TRÊS MARIAS | UNINASSAU | UFPB | UNIFE |

Fonte: Autor

- Tempo de percurso: 116 minutos

TABELA 7: ROTA BR230 COM CRONOMETRAGEM DO PERCURSO

| | | | | | |
|----------|-----|-------------|-----------|------|-------|
| 0 | 4 | 5 | 3 | 2 | 1 |
| TIMBAÚBA | FPB | TRÊS MARIAS | UNINASSAU | UFPB | UNIFE |

Fonte: Autor

- Tempo de percurso: 119 minutos

TABELA 8: ROTA BR101 COM REFERÊNCIA DO GOOGLE MAPS

| | | | | | |
|----------|-------|------|-----------|-----|-------------|
| 0 | 1 | 2 | 3 | 4 | 5 |
| TIMBAÚBA | UNIFE | UFPB | UNINASSAU | FPB | TRÊS MARIAS |

Fonte: Autor

- Tempo de percurso: 128 minutos

TABELA 9: ROTA BR101 COM CRONOMETRAGEM DO PERCURSO

| | | | | | |
|----------|-------|-----|-------------|-----------|------|
| 0 | 1 | 4 | 5 | 3 | 2 |
| TIMBAÚBA | UNIFE | FPB | TRÊS MARIAS | UNINASSAU | UFPB |

Fonte: Autor

Neste caso, ocorreu uma sub-rota que foi [0-1-0] (Timbaúba-UNIPÊ-Timbaúba), portanto implementou-se um caminho alternativo da modulação DFJ (1954) que busca eliminá-la através da restrição que segue abaixo:

Onde, elimina-se a visita repetitiva em um determinado local X_{ij} ou X_{ji} , informando que é necessário visitar uma instância apenas uma única vez.

$$X_{ij} - X_{ji} \leq 1 \quad ; \quad \forall_{ij} \forall_{ji} \in N$$

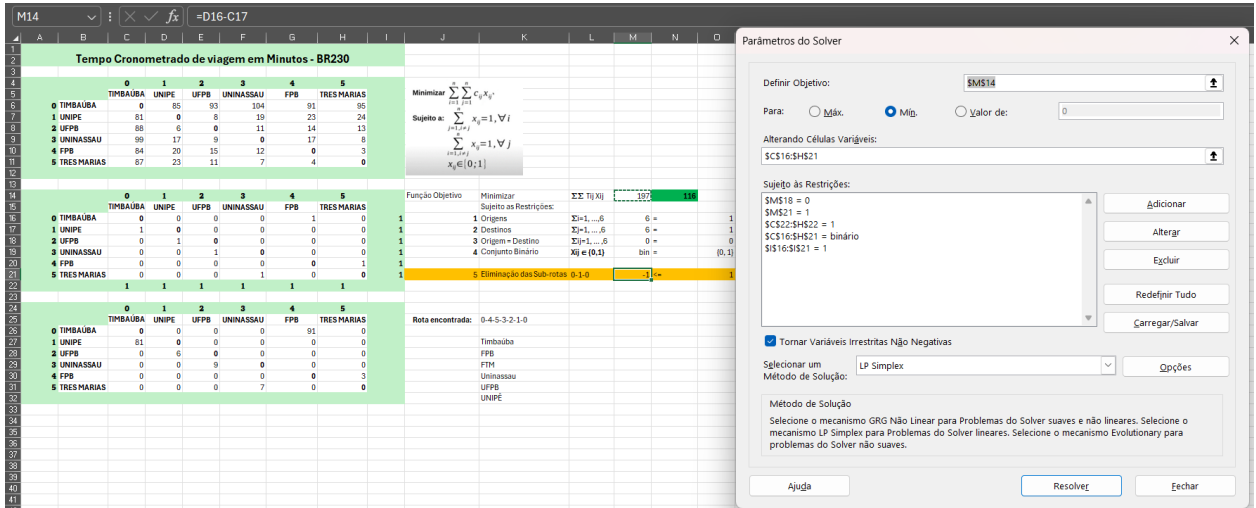
$$X_{ij} X_{ji} = bin$$

Para eliminar o sub-caminho [0-1-0] = Timbaúba-UNIPÊ-Timbaúba, portanto, calcula-se:

$$X_{12} - X_{21} \leq 1 \quad ; \quad \forall_{ij} \forall_{ji} \in N$$

Portanto, ao observar os resultados identifica-se que ambos os cálculos, com decisão de viagem, pela BR230 obtiveram um menor tempo de percurso com diferença apenas de um minuto entre elas (114 minutos e 116 minutos respectivamente), também indicaram uma rota igual, ou seja, o percurso [0, 4, 5, 3, 2, 1] representado pelo grafo abaixo em seguida.

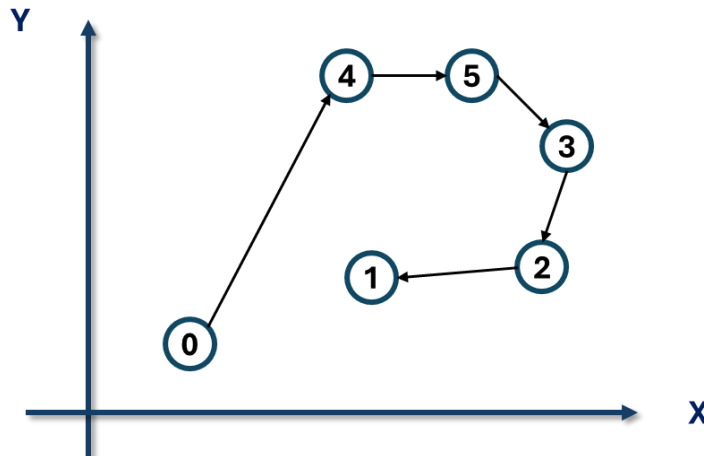
FIGURA 5: MODULAÇÃO DFJ (1954) COM ALTERNATIVA DE ELIMINAÇÃO DE SUBROTAS ATRAVÉS DO MÉTODO SIMPLEX.



Fonte: Autor.

A Figura 5 representa a construção do modelo DFJ (1954) em excel utilizando cálculo de eliminação de sub-caminhos ou sub-rotas como explicado anteriormente.

FIGURA 6: Grafo da rota de ida ideal encontrada pelo solver.



Fonte: Autor

A rota pela BR230, portanto, será a selecionada para aplicação nos próximos métodos e enfim calcular e comparar os resultados. Como identifica-se que tanto pelo Google Maps quanto pela cronometragem o resultado foi homogêneo, utilizando da

matriz de tempo cronometrada (Tabela 2) para que a solução seja a mais adequada e real possível.

4.3.1. PYTHON: UTILIZAÇÃO DA LINGUAGEM PARA UMA SOLUÇÃO MAIS ADEQUADA

Nesta seção serão propostos e apresentados duas heurísticas diferentes para solução do PCV, a heurística do vizinho mais próximo e um código genético com base na modelação MTZ e utilização de um pacote de otimização denominado Gurobi. Ambos os códigos das propostas de solução foram desenvolvidos no Google Colab.

A ideia de trazer esses dois modelos é dar mais alternativas de análise nas soluções, identificando os cálculos simples e mais complexos para confrontar as informações e discutir sobre as rotas e tempo de viagem informados, alinhados com o objetivo do trabalho, se será necessário otimizar ou validar as respostas.

4.3.2. HEURÍSTICA DO VIZINHO MAIS PRÓXIMO

A heurística do vizinho mais próximo é uma abordagem simples para resolver o Problema do Caixeiro Viajante (PCV). Nesta heurística, o caixeiro viajante seleciona a cidade mais próxima, ainda não visitada, a partir da cidade atual, e continua selecionando os vizinhos mais próximos até ter visitado todas as cidades e retornar à cidade de origem. Esta é uma estratégia gulosa, pois faz escolhas locais ótimas em cada etapa sem considerar o cenário global.

A heurística desenvolvida foi baseada no Método do Vizinho Mais Próximo (Bellmore e Nemhauser, 1968), onde a partir de um nó arbitrariamente escolhido, escolhe-se o nó com a menor distância para ser o próximo a ser visitado, repetindo o processo até que todos os nós sejam visitados, unindo então o último nó ao inicial.

Abaixo segue um exemplo passo a passo da heurística do vizinho mais próximo:

1. Inicialização: Comece a partir de uma cidade inicial.
2. Seleção do Vizinho Mais Próximo: A partir da cidade atual, escolha a cidade não visitada mais próxima.
3. Visita da Cidade Selecionada: Viaje para a cidade selecionada.
4. Atualização da Lista de Cidades Visitadas: Marque a cidade selecionada como visitada.

5. Verificação de Condição de Parada: Se todas as cidades foram visitadas, retorne à cidade de origem e termine o percurso. Caso contrário, retorne ao passo 2.
6. Retorno à Cidade de Origem: Quando todas as cidades foram visitadas, retorne à cidade de origem para concluir o percurso.

Com esta heurística foi desenvolvido um código em Python para solucionar com maior facilidade e agilidade o problema, onde segue:

```
def vizinho_mais_proximo_matriz(distancias):
    n = len(distancias)

    visitado = [False] * n # cria lista com info de que nenhuma cidade foi visitada

    percurso = [0] # Começa na cidade 0

    visitado[0] = True # cidade 0 (timba) inicia com TRUE pois ela é a origem do caminho

    for _ in range(1, n):
        ultimo = percurso[-1] # código [-1] pega o ultimo item de uma lista

        proximo = None

        distancia_minima = float('inf') # valor infinito (infinitamente grande)

        for i in range(n):
            if not visitado[i] and distancias[ultimo][i] < distancia_minima:
                proximo = i
                distancia_minima = distancias[ultimo][i]

        percurso.append(proximo)
        visitado[proximo] = True

    percurso.append(0) # Retorna à cidade de origem ao final do percurso

    return percurso

# Matriz de distâncias entre os ponto
```

```
distancias = [
    [0, 85, 93, 104, 91, 95],
    [81, 0, 8, 19, 23, 24],
    [88, 6, 0, 11, 14, 13],
    [99, 17, 9, 0, 17, 8],
    [84, 20, 15, 12, 0, 3],
    [87, 23, 11, 7, 4, 0]
]

percurso = vizinho_mais_proximo_matriz(distancias)

print("Percurso:", percurso)

# Este código usa a heurística do vizinho mais próximo, o que significa
que
# ele procura a cidade não visitada mais próxima para ser a próxima
parada,
# até que todas as cidades tenham sido visitadas, e então retorna à
origem.

# -----

# 0 -> Timbaúba
# 1 -> Unipe
# 2 -> UFPB
# 3 -> Uninassau
# 4 -> FPB
# 5 -> Tres Marias
```

```
#-----
# Para calcular o tempo de percuso soma-se os tempos Xij de origem
# ao destino Xij informado pela rota definida acima

tempo_de_percurso = 85+8+11+8+4

print("tempo_de_percurso:", tempo_de_percurso)

Percurso: [0, 1, 2, 3, 5, 4]
tempo de percurso: 116
```

Melhor opção de percurso: [0, 1, 2, 3, 5, 4] com tempo de 116 minutos. Este grafo também é representado pelo grafo da Figura 3: Grafo da rota de ida comum realizada, na seção 4.2 Coleta de dados.

Este percurso valida o realizado atualmente pelo transporte universitário que obtém os mesmos resultados. Isso indica que o percurso comumente realizado pode ser sim uma ótima opção de rota com um bom aproveitamento do tempo de viagem.

Este é um exemplo simplificado da heurística do vizinho mais próximo. Embora essa abordagem seja fácil de implementar e computacionalmente eficiente, ela não garante uma solução ótima para o PCV e pode resultar em soluções sub-ótimas, especialmente para instâncias grandes e complexas do problema.

Nota-se que esta solução traz um percurso que apenas se inverte da solução pela modelagem no excel e calculado pelo solver, isso significa um certo padrão nas soluções indicando novamente uma sequência similar.

Buscando uma validação desta rota com maior credibilidade, foi desenvolvido outro código em Python, dessa vez através do Gurobi, uma ferramenta complementar do Python. Nesta tentativa, é necessário avaliar a sua solução e confrontar com o método do vizinho mais próximo para que se possa analisar de fato que um algoritmo mais robusto consegue informar uma possibilidade mais coerente e confiável.

4.3.3. APLICAÇÃO DO MÉTODO MTZ EM LINGUAGEM DE PROGRAMAÇÃO PYTHON.

Nesta solução um código genético é utilizado na construção do problema, adotando a modelagem do problema MTZ (Miller, Tucker e Zemlin, 1960).

É mais robusto e imersivo, possui maior complexidade e demonstra com maior exatidão o percurso ideal. Portanto:

Minimizar:

$$\sum_{i=1}^n \sum_{j=1}^n C_{ij} X_{ij} \quad (1)$$

Sujeito a:

$$\sum_{i=1, i \neq j}^n x_{ij} = 1, \quad \forall j \in N \quad (2)$$

$$\sum_{j=1, j \neq i}^n x_{ij} = 1, \quad \forall i \in N \quad (3)$$

$$X_{ij} \in \{0; 1\} \quad (4)$$

Eliminando subcaminhos:

Com programação dinâmica e algoritmos genéticos, para eliminar subrotas, garante-se que a quantidade de rotas possíveis para cada combinação de locais seja menor ou igual a $u - 1$, onde u é a quantidade de locais combinados. Isso evita o fechamento de sub-rotas.

$$u_i - u_j + nx_{ij} \leq n - 1, \quad 2 \leq i \neq j \leq n$$

$$u_i \leq n - 1, \quad \forall i = 2, \dots, n$$

$$x_{ij} \in \{0; 1\}$$

$$u_i \in N_0, \quad i = 2, \dots, n$$

Em seguida se aplica esses conceitos dentro do Google Colab em linguagem Python.

Segue código em Python com auxílio da ferramenta completo de programação Gurobi.

```
import gurobipy as gp
# Parâmetros do problema
qtd_pontos = 6
# Matriz de custos
mat_custos = [
    [ 0, 85, 93, 104, 91, 95],
    [81, 0, 8, 19, 23, 24],
    [88, 6, 0, 11, 14, 13],
    [99, 17, 9, 0, 17, 8],
    [84, 20, 15, 12, 0, 3],
    [87, 23, 11, 7, 4, 0]
]

# Índices dos pontos de origem e destino
origens = [i + 0 for i in range(qtd_pontos)]
destinos = [i + 0 for i in range(qtd_pontos)]

# Dicionário dos custos
custos = dict()
for i, origem in enumerate(origens):
    for j, destino in enumerate(destinos):
        custos[origem, destino] = mat_custos[i][j]

origens
[0, 1, 2, 3, 4, 5]

# Inicializa o modelo
m = gp.Model()

# Variáveis de decisão
x = m.addVars(origens, destinos, vtype=gp.GRB.BINARY)
u = m.addVars(origens[1:], vtype=gp.GRB.INTEGER, ub=qtd_pontos - 1)

# Função Objetivo
m.setObjective(x.prod(custos), sense=gp.GRB.MINIMIZE)

# Restrições que garantem que cada ponto será origem exatamente uma vez
c1 = m.addConstrs(
    gp.quicksum(x[i, j] for j in destinos if i != j) == 1
    for i in origens)

# Restrições que garantem que cada ponto será destino exatamente uma vez
c2 = m.addConstrs(
    gp.quicksum(x[i, j] for i in origens if i != j) == 1
    for j in destinos)
```

```
# Restrições de eliminação de subrotas
c3 = m.addConstrs(
    u[i] - u[j] + qtd_pontos * x[i, j] <= qtd_pontos - 1
    for i in origens[1:] for j in destinos[1:] if i != j)

# Executa o modelo
m.optimize()
```

```
Gurobi Optimizer version 11.0.1 build v11.0.1rc0 (win64 - Windows 11.0
(22631.2))
```

```
CPU model: AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx, instruction set
[SSE2|AVX|AVX2]
Thread count: 4 physical cores, 8 logical processors, using up to 8
threads
```

```
Optimize a model with 32 rows, 41 columns and 120 nonzeros
Model fingerprint: 0xc9896efa
```

```
Variable types: 0 continuous, 41 integer (36 binary)
```

```
Coefficient statistics:
```

```
Matrix range [1e+00, 6e+00]
```

```
Objective range [3e+00, 1e+02]
```

```
Bounds range [1e+00, 5e+00]
```

```
RHS range [1e+00, 5e+00]
```

```
Found heuristic solution: objective 232.0000000
```

```
Presolve removed 0 rows and 6 columns
```

```
Presolve time: 0.01s
```

```
Presolved: 32 rows, 35 columns, 120 nonzeros
```

```
Variable types: 0 continuous, 35 integer (30 binary)
```

```
Root relaxation: objective 1.943333e+02, 13 iterations, 0.00 seconds (0.00
work units)
```

| Nodes | | Current Node | | | Objective Bounds | | | Work | |
|-------|--------|--------------|-------|--------|------------------|--------|-----|---------|------|
| Expl | Unexpl | Obj | Depth | IntInf | Incumbent | BestBd | Gap | It/Node | |
| | | | | | | | | | Time |

| | | | | | | | | | | |
|---|---|---|------------|---|---|--------------|------------|-------|---|----|
| | 0 | 0 | 194.333333 | 0 | 6 | 232.000000 | 194.333333 | 16.2% | - | 0s |
| H | 0 | 0 | | | | 219.00000000 | 194.333333 | 11.3% | - | 0s |
| H | 0 | 0 | | | | 200.00000000 | 194.333333 | 2.83% | - | 0s |
| H | 0 | 0 | | | | 197.00000000 | 194.333333 | 1.35% | - | 0s |
| | 0 | 0 | 194.333333 | 0 | 6 | 197.000000 | 194.333333 | 1.35% | - | 0s |

```
Explored 1 nodes (13 simplex iterations) in 0.08 seconds (0.00 work units)
Thread count was 8 (of 8 available processors)
```

```
Solution count 4: 197 200 219 232
```

```
Optimal solution found (tolerance 1.00e-04)
```

```
Best objective 1.97000000000000e+02, best bound 1.97000000000000e+02, gap
0.0000%
```

```
# Constrói o vetor com o circuito
```

```
circuito = [0]
```

```
anterior = 0
```

```

for ponto in range(qtd_pontos):
    for j in destinos:
        if round(x[anterior, j].X) == 1:
            circuito.append(j)
            anterior = j
            break

# Imprime o circuito
print(circuito)
[0, 4, 5, 3, 2, 1, 0]

# Constrói o vetor com o circuito
circuito = [0]
anterior = 0
for ponto in range(qtd_pontos):
    for j in destinos:
        if round(x[anterior, j].X) == 1:
            circuito.append(j)
            anterior = j
            break

# Remove a última entrada do circuito
if len(circuito) > 1 and circuito[0] == circuito[-1]:
    circuito = circuito[:-1]

print(circuito)
[0, 4, 5, 3, 2, 1]

# Imprime a matriz de valores da variável X
for i in origens:
    print("{:02d}: ".format(i), end="")
    for j in destinos:
        print(round(x[i, j].X), "", end="")
    print("")
00: 0 0 0 0 1 0
01: 1 0 0 0 0 0
02: 0 1 0 0 0 0
03: 0 0 1 0 0 0
04: 0 0 0 0 0 1
05: 0 0 0 1 0 0

```

O grafo da rota encontrada é idêntico ao grafo da Figura 6: Grafo da rota de ida ideal encontrada pelo solver.

Nesse método, assim como nos anteriores, subtrai-se a última entrada do circuito, ou seja, retorno imediato para origem, informando que este veículo aguarda no ponto 1 (UNIPÊ) até iniciar o recolhimento dos estudantes novamente realizando o percurso inverso e retorno, enfim, para Timbaúba-PE (ponto 0).

4.4. CONFRONTO DAS SOLUÇÕES OBTIDAS: COMPARAÇÃO DOS TRAJETOS OBTIDOS PELAS SOLUÇÕES E SELEÇÃO DO MAIS ADEQUADO

Há três possibilidades com três modelos heterogêneos, será comparado as soluções e identificar a similaridade entre elas, tanto para ida quanto no retorno.

TABELA 10 - MÉTODOS UTILIZADOS E SOLUÇÕES OBTIDAS

Fonte: Autor.

| ROTA | SENTIDO | FERRAMENTA | MODELAGEM | MÉTODO | TEMPO DE PERCURSO (FUN. OBJ.) | GRAFO DE ROTA | PERCURSO DEFINIDO |
|------|---------|------------|-----------------------|------------|-------------------------------|---------------|--|
| 1 | IDA | EXCEL | DFJ | SOLVER | 116 MIN | [0-4-5-3-2-1] | TIMBAÚBA-FPB-TRÊS MARIAS-UNINASSAU-UFPB-UNIFE |
| 2 | IDA | PYTHON | VIZINHO MAIS PRÓXIMO | HEURÍSTICA | 116 MIN | [0-1-2-3-5-4] | TIMBAÚBA-UNIFE-UFPB-UNINASSAU-TRÊS MARIAS-FPB |
| 3 | IDA | PYTHON | MTZ - CÓDIGO GENÉTICO | HEURÍSTICA | 116 MIN | [0-4-5-3-2-1] | TIMBAÚBA-FPB-TRÊS MARIAS-UNINASSAU-UFPB-UNIFE |
| 1 | RETORNO | EXCEL | DFJ | SOLVER | 122 | [1-2-3-5-4-0] | UNIFE-UFPB-UNINASSAU-TRÊS MARIAS-FPB- TIMBAÚBA |
| 2 | RETORNO | PYTHON | VIZINHO MAIS PRÓXIMO | HEURÍSTICA | 106 | [4-5-3-2-1-0] | FPB-TRÊS MARIAS-UNINASSAU-UFPB-UNIFE- TIMBAÚBA. |
| 3 | RETORNO | PYTHON | MTZ - CÓDIGO GENÉTICO | HEURÍSTICA | 122 | [1-2-3-5-4-0] | UNIFE-UFPB-UNINASSAU-TRÊS MARIAS-FPB- TIMBAÚBA |

Nota-se grande familiaridade entre as três possibilidades resultantes de modelagem e métodos diferentes com a rota já realizada atualmente, os três métodos entregam o mesmo tempo de percurso na ida. Os cálculos nos mostram duas possibilidades de rotas.

Primeiramente, percebe-se que a heurística do vizinho mais próximo reafirma a rota comum já realizada pelo ônibus, cronometrada de acordo com o tópico 4.2 deste trabalho.

Segundamente, é nítido que o cálculo pelo solver mostra um resultado igual quando comparado a modelagem em Python pelo código genético da heurística MTZ.

Evidenciando, agora, o resultado da função objetivo do problema, ou seja, a minimização do tempo de percurso, entende-se que a diferença entre os resultados são mínimas com apenas três minutos para menos nos cálculos de rota realizados no Excel e na modelagem Python 2.

É notável que pela mínima diferença entre os resultados as três opções de rotas podem ser escolhidas. Inclusive, as rotas são o inverso da outra o que não exclui, por exemplo, a decisão de ir da origem até a FPB ou da origem até a UNIPÊ e seguir o fluxo, pois ambas são início e final de percurso, complementares.

Para comprovar a validade dos percursos, as tabelas abaixo comprovam que na escolha da rota 1 ou rota 2 os alunos desembarcam nas suas instituições antes das 18hrs da tarde, evidenciando que estão dentro do horário, pois o início das aulas do período noturno varia de faculdade para faculdade, onde algumas iniciam o noturno às 18hrs e outras até as 19hrs.

Por fim, sobre o retorno para Timbaúba-PE após o término do horário de aulas, entende-se que a rota comum é de fato a melhor alternativa pois entrega uma diferença média em torno de 16 minutos economizando tempo na chegada em Timbaúba-PE atendendo as restrições de saída de João Pessoa-PB e chegada em Timbaúba-PE por volta das 00hrs ou antes.

TABELA 11 – Horário de chegada e desembarque através da rota comum

| Rota | Instituição | Desembarque |
|-------------|--------------------|--------------------|
| 0 | Timbaúba | 16:00:00 |
| 1 | UNIPE | 17:25:00 |
| 2 | UFPB | 17:33:00 |
| 3 | UNINASSAU | 17:44:00 |

| | | |
|---|-------------|----------|
| 5 | Três Marias | 17:52:00 |
| 4 | FPB | 17:56:00 |

Fonte: Autor.

TABELA 12 – Horário de chegada e desembarque através da rota mensurada.

| Rota | Instituição | Desembarque |
|------|-------------|-------------|
| 0 | Timbaúba | 16:00:00 |
| 4 | FPB | 17:31:00 |
| 5 | Três Marias | 17:34:00 |
| 3 | Uninassau | 17:41:00 |
| 2 | UFPB | 17:50:00 |
| 1 | UNIPE | 17:56:00 |

Fonte: Autor.

TABELA 13 – Horário de embarque para retorno através da rota comum.

| Rota | Instituição | Desembarque |
|------|-------------|-------------|
| 4 | FPB | 21:50:00 |
| 5 | Três Marias | 21:53:00 |
| 3 | Uninassau | 22:00:00 |
| 2 | UFPB | 22:09:00 |
| 1 | UNIPE | 22:15:00 |
| 0 | Timbaúba | 23:36:00 |

Fonte: Autor.

TABELA 14 – Horário de embarque para retorno através da rota mensurada.

| Rota | Instituição | Desembarque |
|------|-------------|-------------|
| 1 | UNIPE | 21:50:00 |
| 2 | UFPB | 21:58:00 |
| 3 | UNINASSAU | 22:09:00 |
| 5 | Três Marias | 22:17:00 |
| 4 | FPB | 22:28:00 |
| 0 | Timbaúba | 23:52:00 |

Fonte: Autor.

Em suma, é visível que a rota comum atende a todos os critérios estabelecidos e de fato é a melhor alternativa para minimização dos tempos de percurso e, conseqüentemente, na redução de custos do transporte universitário.

5. CONSIDERAÇÕES FINAIS

O trabalho mostra três possibilidades diferentes para obtenção de um resultado simples, entretanto para problemas de maior complexidade e número de variáveis (locais) o modelo mais adequado é a modelagem em Python com o código genético visto que é uma modelagem que irá calcular com maior eficiência as interações de local a local, uma vez que o PCV, por ser um problema NP-difícil necessita de uma robustez maior para o cálculo de milhares de interações se necessário, o que não é o caso desta problemática que possui seis instâncias.

Também se entende que a rota atual é de fato a ideal, quando comparado os resultados nas rotas de ida e de retorno o qual obteve-se uma maior economia de em média 16 minutos no tempo de viagem, sendo este resultando totalmente preponderante para essa análise e definição, visto que os resultados das rotas de ida se assemelham muito.

É visível que a heurística do vizinho mais próximo atende de maneira muito direta ao objetivo geral do trabalho, mesmo sendo mais simples que os outros dois métodos discutidos, mostrando que para este tipo de situação pode ser considerada e validada.

O trabalho entregou, então, a validação da rota já percorrida pelo transporte universitário após analisar as rotas resultantes dos cálculos no trabalho, sem excluir a possibilidade de otimização para a rota 1 (Tabela 10) numa margem de erro de cinco minutos para cima e para baixo.

A intenção deste material é poder evidenciar que problemas simples podem ser solucionados através de cálculos menos complexos e de formas e métodos diferentes para se alcançar um resultado satisfatório.

Além disso, é preciso deixar nítido que um problema de ordem pública do município de Timbaúba-PE foi analisado, estudado, validado e otimizado pela universidade pública através de um discente de engenharia filho dessa terra, que enfrentou por dez anos os desafios e adversidades deste transporte na luta pela conclusão do ensino superior provando que é possível e mais que necessário a união dos laços entre a universidade e a sociedade em busca do desenvolvimento mútuo com empenho e apoio o educacional fundamental.

Muitas informações, que poderiam deixar este documento ainda mais robusto com mais detalhes e abordar, quem sabe, outras problemáticas, foram negligenciadas.

É de suma importância o apoio da comunidade acadêmica e dos usuários deste transporte, pois contribuem para a melhoria do seu dia a dia. Após a apuração das informações necessárias e investigadas é que se torna possível compreender a forte necessidade e influência deste trabalho para a sociedade.

Este trabalho deverá ser utilizado como referência para construção de outros modelos e solução de outras problemáticas mais complexas em que o PCV possa ser fundamental. Ele dá embasamento para diversas áreas na sociedade, essencialmente na área logística que enfrenta todos os dias desafios de complexidades muito variáveis. Há três modelos para cálculos de rotas em que outros trabalhos, artigos tenham base e utilizem-no sem moderação.

Por fim, é de grande importância contribuir para a comunidade acadêmica, científica e universitária. É motivo de orgulho poder dissertar sobre conceitos tão atuais e resolutivos. O aprendizado é constante, a cada letra digitada, a cada pesquisa realizada, a cada artigo indicado pela excepcional orientadora... importante mesmo é entender que a educação é uma via apenas de ganhos, com ela não há o que perder. Alguns desafios, enfrentados para elucidação deste trabalho de conclusão de curso, foram enfrentados com muita coragem e persistência com a finalidade de entregar algo de extrema importância e utilidade, que encha o leitor de conhecimento e consiga entregar uma didática simples para que todas as pessoas possam compreender o que aqui está documentado.

Toda construção educacional é válida. Conhecimento não ocupa espaço.

6. REFERÊNCIAS BIBLIOGRÁFICAS

Algoritmos Genéticos. Disponível em: <https://sites.icmc.usp.br/andre/research/genetic/>. Acesso em 27 de mar. 2024.

ARENALES, Marcos Nereu et al. **Pesquisa operacional para cursos de engenharia.** 1 ed. Rio de Janeiro: Elsevier, 2007. 544 p.

ARENALES, Marcos; AMENATO, Vinícius; MORABITO, Reinaldo; YANASSE, Horacio. **Pesquisa Operacional para cursos de engenharia.** [Rio de Janeiro-RJ]: Editoras Campus e Elsevier, ABEPRO, 2011.

BELLMORE, Mandell; NEMHAUSER, George L. The traveling salesman problem: a survey. **Operations Research**, v. 16, n. 3, p. 538-558, 1968.

BEZERRA, F. Participação social no PPA 2016-2019: limites e possibilidades da democratização do processo político-decisório no planejamento e orçamento governamental. In: **Brasília: X Congresso CONSAD de Gestão Pública.** 2017.

BRASIL, A. **Curiosidades sobre a malha rodoviária brasileira.** Disponível em: <https://www.avepbrasil.com.br/curiosidades-sobre-a-malha-rodoviaria-brasileira>. Acesso em: 2 maio. 2024.

BOWERSOX, D. J.; CLOSS, D.; COOPER, M. **Gestão da cadeia de suprimentos e logística.** Rio de Janeiro: Elsevier, 2007

CADERNO de informações técnicas. Ônibus rural escolar. FNDE 2013. Disponível em: https://www.fnde.gov.br/phocadownload/programas/caminho_da_escola/manuais/caderno%20de%20informaes%20tcnicas%20do%20onibus%20escolar%20rural.pdf. Acesso em: 11 de abril 2024.

CAXITO, Fabiano et al. **Logística: um enfoque prático.** 3 ed. São Paulo: Saraiva, 2019. E-book. 376 p. ISBN 9788571440043. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788571440043/>. Acesso em: 28 mar. 2024.

CHRISTOPHER, Martin (Ed.). Logistics: The strategic issues. **London: Chapman & Hall**, 1992.

CONTE, N. **O problema do caixeiro viajante, teoria e aplicações.** 2002. 99 f. Dissertação (Mestrado) - Curso de Matemática Aplicada, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2002. Disponível em: <https://lume.ufrgs.br/bitstream/handle/10183/118198/000339835.pdf?sequence=1&isAllowed=y>. Acesso em: 18 abr. 2024.

CORMEN, Thomas H. et al. Introduction to algorithms. **Cambridge, Massachusett: The MIT Press**, 2022. Disponível em: https://www.researchgate.net/publication/235439156_Handbook_of_meta-heuristics. Acesso em: 08 mai. 2024.

DANTZIG, G; FULKERSON, R; JOHNSON, S. Solution of a large-scale travelingsalesman problem. **Journal of the operations research society of America**, v. 2, n. 4, p. 393-410, 1954

Dias, Marco A. **Introdução à Logística - Fundamentos, Práticas e Integração**. Disponível em: Minha Biblioteca, Grupo GEN, 2016.

GLOVER, F. **Handbook of metaheuristics**. Boston: Kluwer Academic Publishers, 2003.

INSTITUTO SEMESP. **Dados Estados e Regiões**. 13 ed. São Paulo, 2023. Disponível em: <<https://www.semesp.org.br/mapa/edicao-13/regioes/>>. Acesso em: 8 abr. 2024.

KAHNEMAN, D.; TVERSKY, A. **Judgment under uncertainty : heuristics and biases**. Cambridge: Cambridge University Press, 1974.

KNUTH, Donald Ervin. The art of computer programming. **Pearson Education**. 1 ed. vol. 4, fasc. 3. Addison-Wesley Professional, 1997. 160p.

LACHTERMACHER, Gerson. **Pesquisa Operacional na tomada de decisões**. 5 ed. Rio de Janeiro-RJ: Grupo GEN, 2016. E-book. 204 p. ISBN 9788521630494. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788521630494/>. Acesso em: 07 mar. 2024.

LAUDARES, H. Crise de confiança, renovação da política brasileira. **Nexo Jornal**, 21 ago. 2016. Disponível em: <<https://www.nexojornal.com.br/ensaio/2016/08/21/crise-de-confianca-renovacao-da-politica-brasileira>>. Acesso em: 5 abr. 2024.

LONGARAY, André A. **Introdução à Pesquisa Operacional**. São Paulo-SP: Editora Saraiva, 2013. E-book. 285 p. ISBN 9788502210844. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788502210844/>. Acesso em: 07 mar. 2024.

LUTZ, Mark; ASCHER, David. **Aprendendo Python**. 2ª Edição. Porto Alegre: Bookman, 2007. Disponível em: www.facom.ufu.br/~william/Disciplinas%202019-1/BIOTCH-GBT017-IntoducaoInformatica/285173966-aprendendo-python-pdf.pdf. Acesso em: 31 mar. 2024.

MILLER, C. E.; TUCKER, A. W.; ZEMLIN, R. A. **Integer Programming Formulation of Traveling Salesman Problems**. *Journal of the ACM*, v. 7, n. 4, p. 326–329, out. 1960. Disponível em: [chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://dl.acm.org/doi/pdf/10.1145/321043.321046](https://dl.acm.org/doi/pdf/10.1145/321043.321046). Acesso em: 03 mar. 2024.

MITCHELL, M. **Complexity**. [s.l.] Oxford University Press, 2009.

MOREIRA, Daniel A. **Pesquisa Operacional: Curso Introdutório**. São Paulo-SP: Cengage Learning Brasil, 2018. E-book. 376 p. ISBN 9788522128068. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788522128068/>. Acesso em: 07 mar. 2024.

NP (complexidade). Disponível em: [https://pt.wikipedia.org/wiki/NP_\(complexidade\)](https://pt.wikipedia.org/wiki/NP_(complexidade))>. Acesso em: 24 abr. 2024.

Pesquisa Operacional II - **Aula 6: Otimização em Redes (Parte 1/2)**. Disponível em: <https://www.youtube.com/watch?v=uBCpeVnZ4Gw>>. Acesso em: 2 maio. 2024.

PILGRIM, Mark. **Dive Into Python 3**. Apress, 2010. E-book. 360 p. ISBN-13 978-1430224150. Disponível em: xxxxxxxxxx Acesso em: xxxx
 PITOMBEIRA-NETO, Anselmo Ramalho. **Problema do Caixeiro Viajante - Modelagem por programação linear inteira**. 2021. 1 vídeo (49 min. e 47segs.). Disponível em: <https://www.youtube.com/watch?v=53WNmUldPk>>. Acesso em: 20 abr. 2024.

PÓLYAG.; CONWAY, J. H. **How to solve it : a new aspect of mathematical method**. Princeton N.J.: Princeton University Press, 2004.

RAZZOLINI, Edelvino Filho. **Transporte e modais: com suporte de TI e SI**. [s.l.] Editora Ibpex, 2007.

RUSSEL, Stuart; NORVIG, Peter. **Inteligência Artificial**. Tradução da Terceira Edição. Rio de Janeiro, RJ: Elsevier Editora, 2013.

ROBERTS, Eric S.; KELLEY, Al; POHL, Ira. **The Art and Science of C. Laboratório**, v. 50, p. 19, 1995.

SEDGEWICK, Robert. **Algorithms in C, Parts 1-4: Fundamentals, Data Structures, Sorting, Searching**. 3 ed. Pearson Education Limited, 1998. 720 p.

SIMON, H. A. **The sciences of the artificial**. Cambridge, Massachusetts: The Mit Press, 1969.

SOBRAPO - Sociedade Brasileira de Pesquisa Operacional. **O que é a Pesquisa Operacional?** São Paulo, 2014. Disponível em: <https://www.sobrapo.org.br/o-que-e-pesquisa-operacional>. Acesso em: Março de 2024.

SWEIGART, A. **Automate the boring stuff with Python : practical programming for total beginners**. San Francisco: No Starch Press, 2015. Disponível em:

https://www.academia.edu/31998064/Automate_the_Boring_Stuff_with_Python_Practical_Programming_for_Total_Beginners_2015_pdf. Acesso em: 21 mar. 2024

2 Modelos de Programação Linear. [s.l: s.n.]. Disponível em: <http://www.deinf.ufma.br/~acmo/grad/PO_c02_v2005.pdf>.