

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

FÁBIO HENRIQUE SOARES DIAS
Orientador: Reinaldo Silva Fortes

**SOLUÇÃO PROBABILÍSTICA PARA MAXIMIZAÇÃO DOS GANHOS
DE MOTORISTAS DE TÁXI E APLICATIVOS**

Ouro Preto, MG
2024

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

FÁBIO HENRIQUE SOARES DIAS

**SOLUÇÃO PROBABILÍSTICA PARA MAXIMIZAÇÃO DOS GANHOS DE
MOTORISTAS DE TÁXI E APLICATIVOS**

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Reinaldo Silva Fortes

Ouro Preto, MG
2024



FOLHA DE APROVAÇÃO

Fábio Henrique Soares Dias

Solução probabilística para maximização dos ganhos de motoristas de táxi e aplicativos

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 9 de Maio de 2025.

Membros da banca

Reinaldo Silva Fortes (Orientador) - Doutor - Universidade Federal de Ouro Preto
Anderson Almeida Ferreira (Examinador) - Doutor - Universidade Federal de Ouro Preto
Bráulio Miranda Veloso (Examinador) - Mestre - PPGCC - UFOP

Reinaldo Silva Fortes, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 9/04/2025.



Documento assinado eletronicamente por **Reinaldo Silva Fortes, PROFESSOR DE MAGISTERIO SUPERIOR**, em 13/05/2025, às 15:50, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0907884** e o código CRC **24F712B6**.

Agradecimentos

Gostaria de agradecer a UFOP pelo ensino público de qualidade, agradeço também ao meu orientador Reinaldo Fortes pelo direcionamento e mentoria. Por fim, mas não menos importante, eu gostaria de agradecer a toda minha família, em especial meus pais, Fábio Nunes e Josefa Ferreira e ao meu irmão Gustavo.

Resumo

Esta monografia tem como objetivo desenvolver um sistema de recomendação de rotas para motoristas de táxi, com foco em aumentar sua rentabilidade, considerando fatores como a probabilidade de encontrar passageiros, o abastecimento e o preço do combustível. Para isso, este trabalho se inspirou em um estudo que recomenda rotas viáveis para táxis elétricos, com o objetivo de aumentar a receita diária do motorista. A partir dessa ideia, realizou-se uma adaptação para um novo contexto, no qual veículos flex foram selecionados como agentes do sistema. Essa escolha permite que o sistema desenvolvido seja aplicável a taxistas que utilizam veículos flex, aproximando-o da realidade. Para isso, preços reais de combustíveis (gasolina e álcool) são considerados no momento do abastecimento do táxi. Visando esse objetivo, com o auxílio do *Simulation of Urban MObility (SUMO)*, foram gerados novos dados de corridas a partir do histórico *Sistema de Posicionamento Global (GPS)* de táxis, contendo informações sobre suas rotas, para que o problema fosse modelado adequadamente. Com base nesses dados, desenvolveu-se o sistema utilizando o algoritmo de Dijkstra e bibliotecas do *SUMO*, com o intuito de auxiliar nos cálculos de probabilidade. Os experimentos realizados mostraram que a comparação entre estratégias de reabastecimento (posto mais barato e mais próximo) evidenciou a vantagem dos postos próximos em cenários nos quais o veículo ainda possui bastante combustível. Além disso, observou-se a eficiência da gasolina em situações em que a quantidade de combustível está mais limitada.

Palavras-chave: Sistema de recomendação. Rotas. Veículos flex. Dados sintéticos.

Abstract

This monograph aims to develop a route recommendation system for taxi drivers, focusing on increasing their profitability by considering factors such as the probability of finding passengers, refueling strategies, and fuel prices. To achieve this, the work was inspired by a study that recommends viable routes for electric taxis, with the goal of increasing the driver's daily revenue. Based on this idea, an adaptation was made to a new context in which flex-fuel vehicles were selected as the system's agents. This choice allows the developed system to be applicable to taxi drivers who use flex-fuel vehicles, bringing it closer to real-world scenarios. For this reason, real fuel prices (gasoline and ethanol) are considered at the time of refueling. With this goal in mind, and using [SUMO](#), new trip data were generated based on the GPS history of taxis, including information about their routes, in order to properly model the problem. Based on these data, the system was developed using Dijkstra's algorithm and SUMO libraries to support probability calculations. The experiments showed that the comparison between refueling strategies (cheapest vs. nearest gas station) highlighted the advantage of nearby stations in scenarios where the vehicle has a high fuel level. Moreover, gasoline proved to be more efficient when the available fuel was limited.

Keywords: Recommendation system. Routes. Flex-fuel vehicles. Synthetic data.

Lista de Ilustrações

Figura 2.1 – Filtragem colaborativa.	6
Figura 2.2 – Filtragem baseada em conteúdo.	7
Figura 2.3 – Hibridização.	7
Figura 2.4 – Interface do OpenStreetMap integrado ao SUMO.	11
Figura 3.1 – Fluxograma do desenvolvimento do método base.	14
Figura 3.2 – Demanda de táxis durante o dia (24 horas).	16
Figura 3.3 – Demanda de táxis por dias na semana.	16
Figura 3.4 – Mapa das áreas de maior demanda.	16
Figura 3.5 – Fluxograma da lógica de decisão dos táxis.	21
Figura 4.1 – <i>Receita Líquida Esperada</i> (RLE) médio por baseline e nível de combustível.	25
Figura 4.2 – Número de corridas e RLE (100% de combustível).	26
Figura 4.3 – Número de corridas e RLE (60% de combustível).	27
Figura 4.4 – RLE médio com 20% de combustível (10L).	28
Figura 4.5 – ENR médio com 50% de combustível (25L).	28

Lista de Tabelas

Tabela 2.1 – Exemplo de matriz de avaliação usuários-itens	5
Tabela 4.1 – Parâmetros da simulação.	23
Tabela 4.2 – Médias das métricas para 10 táxis por baseline (100% de combustível).	24
Tabela 4.3 – Médias das métricas para 10 táxis por baseline (80% de combustível).	25
Tabela 4.4 – Médias das métricas para 10 táxis por baseline (60% de combustível).	26
Tabela 4.5 – Médias de RLE para limites de combustível de 20% e 50%.	27

Lista de Abreviaturas e Siglas

CARS *Sistemas de Recomendação Sensíveis ao Contexto.* 12

GPS *Sistema de Posicionamento Global.* v, 11, 14, 15, 19

LGPD *Lei Geral de Proteção de Dados do Brasil.* 8

MAE *Mean Absolute Error.* 7, 12

MDP *Markov Decision Process.* 2, 11, 17

PDF *Funções de Distribuição de Probabilidade.* 12, 13

RLE *Receita Líquida Esperada.* vii, 8, 17, 24–27, 29

RMSE *Root Mean Square Error.* 7, 12

SR *Sistema de Recomendação.* 1, 4, 6–8, 12

SUMO *Simulation of Urban MObility.* v, vi, 3, 9, 10, 15, 23, 29

TIP *Transporte Individual de Passageiros.* 3

TLR *Time-Location-Relationship.* 11

Sumário

1	Introdução	1
1.1	Justificativa	3
1.2	Objetivos	3
1.3	Organização do Trabalho	3
2	Revisão Bibliográfica	4
2.1	Sistemas de Recomendação	4
2.1.1	Abordagens em Sistemas de Recomendação	4
2.1.2	Métricas de Avaliação	6
2.2	Bases de Dados Sintéticas	8
2.3	Ferramentas Computacionais	9
2.3.1	Simulation of Urban Mobility (SUMO)	9
2.3.2	OpenStreetMap (OSM)	10
2.4	Trabalhos Relacionados	11
3	Desenvolvimento	14
3.1	Coleta e Análise de Dados	14
3.2	Modelagem do Problema	17
3.2.1	Estado do Táxi	17
3.2.2	Ação do Táxi	17
3.2.3	Receita Líquida Esperada	17
3.2.4	Modelo dos Táxis	18
3.3	Desenvolvimento do Sistema	18
3.3.1	Histórico de Corridas	19
3.3.2	Postos de Combustível	20
3.3.3	Funções de Decisão	21
4	Resultados	23
4.1	Parâmetros	23
4.2	Experimentos	24
5	Considerações Finais	29
5.1	Trabalhos Futuros	30
	Referências	31

1 Introdução

O número de pessoas que têm acesso à Internet no mundo vem aumentando significativamente a cada ano, por mais que 33% da população mundial ainda não esteja conectada à internet (UIT, 2023). Mesmo diante desse problema de desigualdade, o volume de informações e dados gerados na internet vem crescendo exponencialmente (UIT, 2015), trazendo consigo alguns benefícios como a facilitação do acesso ao conhecimento, tomada de decisões baseadas em dados, personalização de conteúdos e serviços, e o desenvolvimento de novas tecnologias.

Nesse contexto, os *Sistemas de Recomendação (SRs)* possui um papel importante, visto que eles são eficazes para mitigar a sobrecarga de escolha, filtrando as informações, utilizando dessa filtragem de informações para servir conteúdos e serviços personalizados, que atendam aos gostos dos usuários. Essa personalização, facilitada pelos dados obtidos, possibilita uma experiência agradável e que atenda às necessidades do usuário (MARAN; MARUYAMA; SILVEIRA, 2023). Os sistemas de recomendação têm o papel de prever os itens mais relevantes para os usuários, tendo como base o próprio perfil e preferências do usuário.

Os *SRs* são mecanismos que, por meio de técnicas aprimoradas, buscam analisar e compreender o comportamento dos usuários, para que seja possível fazer recomendações personalizadas de itens ou serviços. O intuito desses sistemas é que o usuário tenha várias possibilidades e que, ainda assim, sejam personalizadas conforme o que ele consome, proporcionando uma boa experiência ao usuário. Tudo isso é possível por meio de algoritmos de filtragem, que podem ser, por exemplo, do tipo baseado em conteúdo, colaborativa ou híbrida.

Partindo para os possíveis campos de aplicação de *SRs*, pode-se dizer que eles são amplamente utilizados em várias áreas da indústria, como *e-commerce*, redes sociais, serviços de entretenimento, saúde, serviços financeiros, entre outros. Um campo que vem sendo muito explorado é o de recomendações de rotas, a exemplo de trabalhos como (YUEN et al., 2019; WAN; GHAZZAI; MASSOUD, 2019; QU et al., 2014; PA; CHAKRABORTY, 2015a; WANG et al., 2018), que em sua maioria possuem um objetivo de atender e otimizar as necessidades de mobilidade urbana. Pensando em ampliar nosso estudo nesse campo, o princípio dessa monografia terá como base a aplicação de um *SR* em ambiente veicular de maneira análoga ao trabalho feito por Tu et al. (2020), que serviu de inspiração para este estudo.

Tu et al. (2020) criam um *SR* de rotas em tempo real para táxis elétricos baseado nas trajetórias GPS de táxis na cidade de Shenzhen, China. O trabalho propõe

um método responsável por ligar dois fatores influentes para as recomendações: (1) a procura por passageiros; e (2) aos pontos de recargas dos veículos elétricos. Para isso, os autores utilizam do *Markov Decision Process (MDP)*, responsável por modelar e otimizar as decisões dos motoristas pensando nas necessidades daquelas ações e nas que poderão maximizar a receita esperada. A partir dessa abordagem, esta monografia visa adaptar esse estudo para a realidade do Brasil, considerando que veículos *flex* são mais populares no país, ao contrário dos veículos elétricos.

Atualmente, no Brasil, os veículos *flex* representam cerca de 85% da frota de veículos leves que circulam no país, além de ser o tipo de automóvel mais produzido no país (ANFAVEA, 2024). Esses tipos de carros possuem um motor de combustão interna e uma de suas funções é permitir que o carro funcione com mais de um tipo de combustível, dos quais, os mais comuns podem consumir tanto gasolina quanto álcool. Essa possibilidade traz uma boa vantagem aos motoristas, visto que, é dado uma maior liberdade no momento do abastecimento, tendo em vista os preços flutuantes da gasolina e do álcool. Outro fator relevante é a diferença de autonomia dos veículos ao utilizar cada um destes combustíveis.

Com toda essa flexibilidade, surgiu a necessidade dos condutores se manterem informados a respeito do preço dos combustíveis, considerando suas constantes variações no cenário brasileiro. Esse propósito de reduzir gastos não é uma realidade apenas para motoristas comuns, mas também para aqueles que utilizam de veículos como forma de sustento, a exemplo dos taxistas. Para esses profissionais a economia de combustível tende a ser um dos principais fatores considerados na escolha do veículo.

Além da influência que o custo dos combustíveis pode ter na otimização do lucro obtido com as corridas, com a grande intensificação e concorrência na área do transporte, surge também a necessidade tomar decisões inteligentes. Essas decisões estão principalmente relacionadas à otimização de rotas, que além de conseguir promover a economia de combustível pode aumentar as chances de encontrar passageiros, possibilitando maiores lucros ao motorista. Logo, um carro *flex* atrelado a boas estratégias de otimização de rotas se mostra promissor no contexto em questão.

Nas seções a seguir apresentamos a justificativa na [Seção 1.1](#) e objetivos na [Seção 1.2](#) do trabalho e como este documento está organizado.

1.1 Justificativa

Diante do contexto apresentado, se mostra válido a busca no domínio da área da recomendação por um sistema treinado que atue na sugestão de rotas que sejam mais promissoras e que possibilitem um maior retorno financeiro para taxistas e motoristas de aplicativo. Considerando que veículo *flex* é o tipo de transporte doméstico mais popular no Brasil, proporcionar um estudo com foco nesse campo poderia impactar e gerar ganhos positivos para um grande grupo de pessoas.

1.2 Objetivos

Este trabalho tem como objetivo principal apresentar um sistema que faça sugestões de rotas para motoristas de *Transporte Individual de Passageiros (TIP)* (táxis ou de aplicativo), visando o aumento da sua rentabilidade, considerando a probabilidade de encontrar passageiros, assim como o desempenho do veículo e o custo de combustível, em especial para veículos *flex*.

Para atingir o objetivo principal, foram selecionado alguns objetivos específicos:

- Desenvolver um novo conjunto de dados sintéticos para realização de experimentos;
- Criar modelos de recomendação de rotas considerando o aumento de receita, os melhores valores de abastecimento de combustível (redução de gastos) e a probabilidade de pegar passageiros;
- Validar o sistema por meio de simulações usando o *Simulation of Urban MObility (SUMO)*.

1.3 Organização do Trabalho

Este trabalho foi dividido em 4 capítulos principais: a Introdução, no [Capítulo 1](#), contextualizando o tema e descrevendo o objetivo principal do estudo; a Revisão Bibliográfica trazendo a fundamentação teórica e os trabalhos relacionados sobre sistemas de recomendação, estratégias para recomendação de rotas e geração de dados sintéticos no [Capítulo 2](#); o Desenvolvimento detalhando a implementação feita no [Capítulo 3](#); os experimentos e análises dos resultados se encontram no [Capítulo 4](#) e por fim, a Conclusão, resumindo as principais conclusões e direcionamentos futuros no [Capítulo 5](#).

2 Revisão Bibliográfica

Este capítulo visa analisar e explorar a literatura de [SRs](#) com o intuito de nos permitir conhecer as principais abordagens e técnicas utilizadas que se encaixem com a proposta deste trabalho. Sendo assim, é feita uma discussão sobre [SRs](#), apresentando conceitos básicos e aplicações na [Seção 2.1](#), métodos para geração de dados sintéticos são discutidos na [Seção 2.2](#) e, na [Seção 2.4](#), são explorados trabalhos com o mesmo escopo desta monografia, ou seja, recomendação de rotas em ambiente veicular.

2.1 Sistemas de Recomendação

O conceito de [SRs](#), é um dos pontos centrais deste trabalho e como dito anteriormente se tornou necessário entender mais sobre o assunto. [SRs](#) são sistemas de filtragem que, a partir de estratégias específicas, conseguirão compreender e aprender os comportamentos e interesses dos usuários para que seja possível realizar previsões que atendam às preferências e às necessidades dos mesmos.

Esses sistemas são classes de algoritmos que recomendam itens relevantes aos usuários esperando que a taxa de aceitação seja a maior possível. As diversas técnicas abordadas para essas recomendações analisam vários fatores, como, por exemplo, os padrões e correlações entre os dados existentes, o histórico de consumo, e interação dos usuários de forma individual ou coletiva ([AZAMBUJA; MORAIS; FILIPE, 2021](#)).

Normalmente, as interações responsáveis por proporcionar as recomendações são representadas por uma matriz usuário-item, onde cada linha vai indicar um usuário e as colunas vão representar um item específico. Essa é uma matriz que pode conter vários tipos de dados, como as avaliações atribuídas àquele item, indicações de compra ou não, interações feitas pelo usuário com o item, entre outras possibilidades. Na [Tabela 2.1](#) temos um exemplo da avaliação de 5 usuários para 5 itens, em que foram atribuídas notas de 1 a 5 para os itens, considerando também a possibilidade do usuário não ter feito nenhuma avaliação ao item.

2.1.1 Abordagens em Sistemas de Recomendação

As abordagens aplicadas nos sistemas fornecem a base para o seu funcionamento, sendo responsáveis por buscar os padrões e comportamentos e aplicá-los em forma de recomendações. As abordagens giram em torno de sistemas persona-

Usuário/Item	Item 1	Item 2	Item 3	Item 4	Item 5
Usuário 1	4	-	2	5	-
Usuário 2	3	5	-	1	2
Usuário 3	-	3	4	-	5
Usuário 4	1	-	3	2	-
Usuário 5	-	2	1	4	3

Tabela 2.1 – Exemplo de matriz de avaliação usuários-itens

lizados e não personalizados. Os sistemas personalizados alisam o comportamento do usuário, suas preferências e interações passadas, já os não personalizados trabalham com dados agregados e regras mais gerais pois não se baseiam em ações específicas do usuário, que é o caso desta monografia tendo em vista que os pesos de demanda, regras de probabilidade e de escolha de postos, por exemplo, são as mesmas para todos os usuários do sistema. As técnicas mais tradicionais para auxiliar nesse processo são as de filtragem colaborativa, filtragem baseada em conteúdo, e a hibridização.

Nesse contexto, a filtragem colaborativa é a abordagem de recomendação mais utilizada, baseando-se nos pontos de interesse comum entre os usuários (CAZELLA; NUNES; REATEGUI, 2010). A sugestão dos itens ocorre a partir da ideia de que usuários com gostos e interesses semelhantes vão ter uma tendência a consumirem as mesmas coisas. Dessa forma, a filtragem colaborativa considera as preferências do usuário e as preferências daqueles usuários que sejam semelhantes a ele, ou seja, ela busca por padrões nas escolhas e condutas do usuário, como ilustrado na [Figura 2.1](#).

Dentre as várias vantagens da filtragem colaborativa podemos destacar a possibilidade de diversificação nos itens recomendados, isso ocorre pois essa abordagem busca por semelhanças nos perfis dos usuários e não no conteúdo dos itens. Uma das principais desvantagens dessa abordagem é o problema do *Cold Start* que ocorre a partir da entrada de novos usuários no sistema e sua dificuldade em gerar recomendações já que ainda não se tem informações sobre esses perfis.

A filtragem baseada em conteúdo traz uma abordagem de realizar análises do conteúdos dos itens e sua relevância para aquele usuário específico. Diferente da filtragem colaborativa que depende das interações de outros usuários para recomendar itens, esta outra técnica priorizará as informações e características de um item para poder encontrar outros que compartilhem de características semelhantes, como pode ser visto na [Figura 2.2](#). Seguindo essa linha, um modelo é criado baseado nas informações a respeito dos itens consumidos pelo usuário e a partir desse modelo comparações entre os atributos do perfil de interesses do

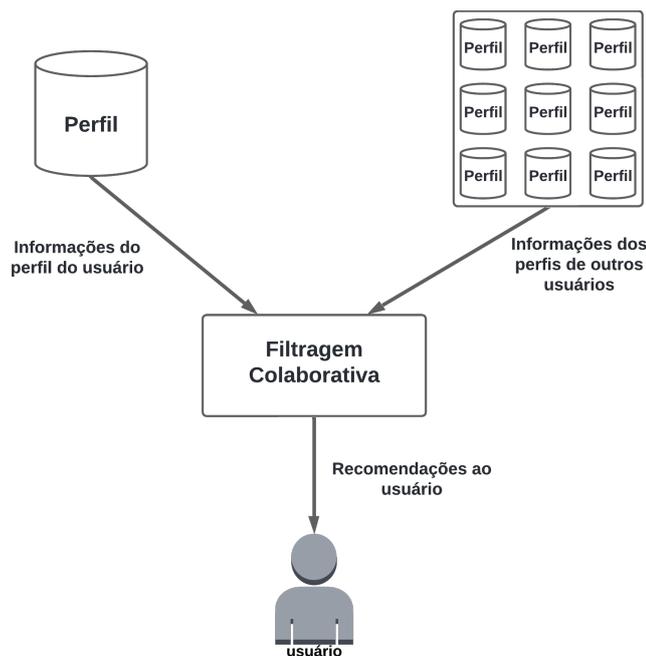


Figura 2.1 – Filtragem colaborativa.

usuário e do novo item são feitas para que se possa realizar as recomendações.

Tratando-se das principais vantagens dessa abordagem, destaca-se o fato dela proporcionar recomendações mais precisas com base apenas no perfil do usuário, além de não depender de outros usuários para realizar as recomendações, o que evita o problema de *Cold Start*. Em contrapartida, há o problema de superespecialização, muito comum nessa abordagem. Isso acontece quando o sistema recomenda itens extremamente parecidos com os que já são consumidos pelo usuário, limitando assim a diversidade das sugestões.

A hibridização é uma abordagem que busca por meio de várias estratégias fazer a combinação de duas ou mais técnicas, como ilustrado na [Figura 2.3](#), para ampliar as vantagens do sistema e para compensar também as desvantagens que alguma outra técnica possa ter. Alguns dos pontos positivos dessa abordagem apresentado por [Cazella, Nunes e Reategui \(2010\)](#) envolvem a possibilidade de descoberta de relacionamentos novos, aumento da precisão das recomendações independente da quantidade de usuários e para usuários incomuns também.

2.1.2 Métricas de Avaliação

As métricas de avaliação são importantes para se ter a garantia da eficácia do [SR](#), essas métricas verificam principalmente a assertividade das recomendações feitas pelo sistema, ou seja, se o usuário está recebendo sugestões relevantes, além disso, outros aspectos podem ser verificados como o nível de diversidade das recomendações. Para fazer essas avaliações, diversas técnicas podem ser

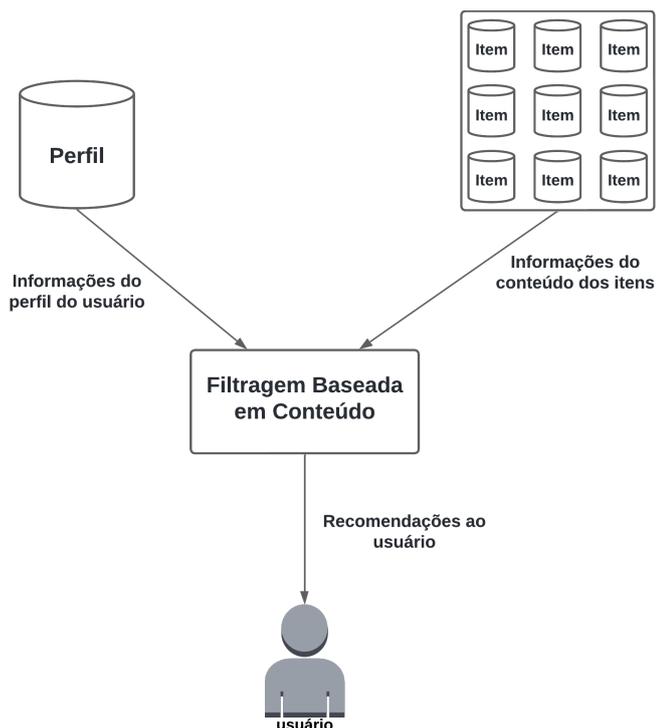


Figura 2.2 – Filtragem baseada em conteúdo.

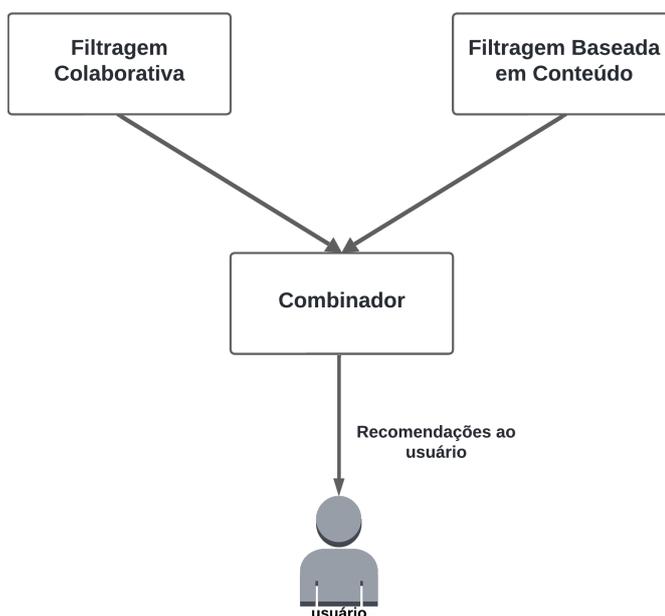


Figura 2.3 – Hibridização.

utilizadas, a depender é claro do tipo de **SR** e do contexto em que está sendo aplicado.

Para **SRs** personalizados, pode-se destacar duas categorias principais de avaliação: métricas de acurácia de predição, a exemplo do *Mean Absolute Error (MAE)*, *Root Mean Square Error (RMSE)*, que avaliam o quão próximo o sistema prevê valores numéricos que usuário daria para um item, como uma nota, por

exemplo, e métricas de acurácia de classificação, como *recall*, *F-measure* que validam a relevância das recomendações feitas (HERLOCKER et al., 2004). Mas, para SR não personalizados algumas métricas específicas, como receita gerada, são mais adequadas pois se alinham com os objetivos do sistema, como utilizado por Tu et al. (2020) com a finalidade de maximizar o lucro.

Para avaliar a eficácia do que foi desenvolvido nesta monografia, utilizou-se como métrica principal a *Receita Líquida Esperada (RLE)*. Essa métrica considera, além da receita obtida com as corridas realizadas, os custos associados à jornada de trabalho, como o gasto de combustível durante o serviço. Ao contrário de métricas mais simples, como a quantidade total de corridas ou a distância percorrida, a RLE oferece uma visão mais assertiva do estudo desenvolvido, pois leva em conta os ganhos em relação à probabilidade de encontrar passageiros e os gastos durante os deslocamentos.

2.2 Bases de Dados Sintéticas

A fim de ajudar este trabalho com o uso e na criação de novos dados para as corridas de táxis, se tornou necessário a pesquisa sobre o tema. À medida que o volume de dados cresce surge a necessidade de que esses dados sejam protegidos, o que se faz necessário regulamentações que garantam a segurança e privacidade dessas informações, já que muitos desses dados são gerados por pessoas físicas, como, por exemplo, ao criar uma conta num banco, num site de compras, entre outras coisas. Por conta dessa privacidade acerca dos dados, os dados sintéticos ganham destaque como uma alternativa que nos permite estudar e manipular essas informações sem precisar lidar com questões regulamentárias como a *Lei Geral de Proteção de Dados do Brasil (LGPD)*.

Diferentemente dos dados reais, os dados sintéticos apresentam a vantagem de poderem ser gerados a partir do mesmo espaço amostral, permitindo a replicação e o uso controlado das informações para fins de análise e estudo. Além disso, possibilitam a realização de comparações diretas quanto à similaridade com os dados originais. Com o objetivo de atender a demandas específicas de análise, os dados sintéticos são produzidos por meio de modelos matemáticos e algoritmos desenvolvidos para essa finalidade. (JORDON et al., 2022).

Os dados sintéticos podem se dividir em alguns tipos como: aqueles completamente gerados onde não são gerados diretamente dos dados reais e criados por modelos estatísticos, simulações ou algoritmos criados para isso, dados de perturbação na qual os dados reais sofrem algum tipo de modificação como ruídos, variações, etc., e os dados semi-sintéticos com uma combinação dos dados reais e

sintéticos.

Uma abordagem bastante comum na geração de dados sintéticos utiliza Funções de Distribuição de Probabilidade. Elas definirão os valores das variáveis aleatórias e seus comportamentos, sendo que esses comportamentos são definidos por nós. Essas funções conseguem modelar diferentes aspectos dos dados que elas podem ser de vários tipos como: distribuição gaussiana, bimodal, exponencial, quiquadrado, etc. A modelagem de dados pode ser feita por variáveis aleatórias simples e suas funções de distribuição de probabilidade e permitem simular o comportamento de usuários em diferentes contextos ([PASINATO et al., 2013](#));

Algumas outras abordagens envolvem a técnica de aumento de dados que é utilizada quando já se tem alguma base real e a partir dela construímos uma nova base de dados mais diversa, robusta e de fácil adaptação para novos exemplos ([IBM, 2024](#)). Essa é uma técnica interessante de geração de dados para situações em que o conjunto de dados real é limitado ou de difícil obtenção.

2.3 Ferramentas Computacionais

Tendo em vista o direcionamento desse trabalho para o ambiente veicular e para a questão de mobilidade urbana, o uso de algumas ferramentas computacionais se tornaram essenciais. Atualmente podemos contar com aplicações que facilitam o estudo de áreas e planejamentos urbanos, que vão desde aplicações mais elaboradas de simulação de tráfegos em qualquer espaço geográfico até uma simples aplicação de geolocalização.

Nesse trabalho são utilizadas ferramentas como o [SUMO](#) para simulação das viagens realizadas pelos táxis apresentado na [Seção 2.3.1](#) e na [Seção 2.3.2](#) temos o OpenStreetMap para obtenção da rede viária de onde o projeto será aplicado.

2.3.1 Simulation of Urban Mobility (SUMO)

O [SUMO](#), é um simulador de tráfego urbano que permite a manipulação de comportamentos de diversos transportes sob diversas circunstâncias. Por conta de sua flexibilidade podemos recriar e editar cenários reais permitindo estudos de comportamentos urbanos que sejam os mais próximos possíveis da realidade. Além disso, essa aplicação nos dá todo um conjunto de ferramentas que possibilita definir rotas, gerar novas viagens, controlar semáforos e até mesmo fazer interações com a simulação em tempo real com a API TraCi controlada por python.

A sua flexibilidade toda, bem como toda a capacidade de integração com diversos formatos de dados, o tornam uma ferramenta essencial para estudos acadêmicos e aplicações práticas relacionadas ao tráfego urbano. Investigações na comunicação veicular (V2X), a otimização das rotas dinâmicas, o controle adaptativo de semáforos, a vigilância do tráfego e o desenvolvimento de algumas estratégias de gestão urbana (BEHRISCH et al., 2011) acham-se entre as aplicações de maior ocorrência.

O SUMO tem sido usado como plataforma de suporte ao desenvolvimento de vários sistemas inteligentes de recomendação de rotas, além de sua estrutura modular e extensível. O SUMO, por exemplo, foi empregado por Tandon, Verma e Gupta (2022) para gerar uma base de dados com mil rotas simuladas, as quais serviram como entrada para modelos de recomendação baseados em previsão sequencial e técnicas de fatoração de matrizes. Comportamentos de tráfego realisticamente reais foram reproduzidos pela simulação, sendo isso algo fundamental para a avaliação da eficácia dos algoritmos propostos em contextos urbanos (TANDON; VERMA; GUPTA, 2022).

Outro uso recorrente do SUMO é como gerador de dados para cenários de tomada de decisão baseada em atribuição dinâmica de tráfego. Nesses contextos, são simuladas tripulações aleatórias de veículos que percorrem a rede urbana, fornecendo dados em tempo real sobre o fluxo e as condições viárias. Tais informações alimentam algoritmos de roteamento que visam minimizar o tempo de viagem e outros custos associados às rotas, contribuindo para decisões mais eficientes em sistemas de transporte inteligentes (PA; CHAKRABORTY, 2015b). Assim, o SUMO configura-se como uma ferramenta robusta e validada para experimentação e simulação de estratégias de mobilidade urbana, especialmente no contexto de desenvolvimento e avaliação de algoritmos de recomendação de rotas.

2.3.2 OpenStreetMap (OSM)

O OpenStreetMap é uma ferramenta de mapeamento coletivo, em que podemos criar e coletar dados geográficos. O seu nível de detalhamento é muito grande, fornecendo dados como ruas, avenidas, rotatórias, linhas de trem, cicloviás e áreas verdes. Para auxiliar neste estudo utilizamos a ferramenta integrada no próprio SUMO, por onde foi extraído todos os dados viários do local desejado com exceção de informações que não agregassem na simulação. Quando é feita a extração da rede viária os dados já são convertidos para serem utilizados e manipulados no SUMO.

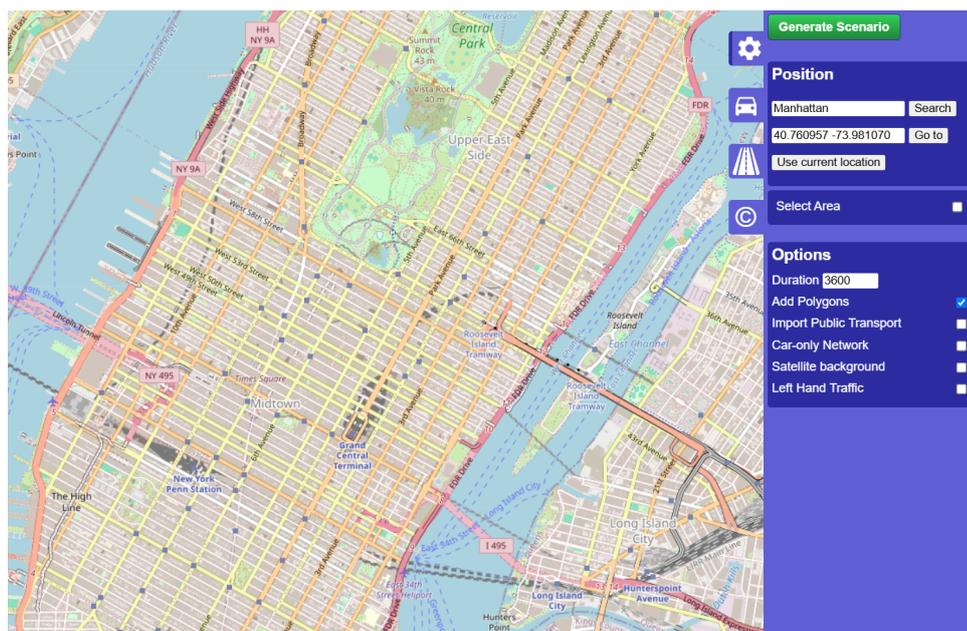


Figura 2.4 – Interface do OpenStreetMap integrado ao SUMO.

2.4 Trabalhos Relacionados

Tu et al. (2020) apresenta o projeto que serviu de inspiração para a escolha do tema desta monografia, compartilhando do mesmo objetivo de maximizar os ganhos dos motoristas. Nele, são aproveitados os dados *Sistemas de Posicionamento Global (GPSs)* dos táxis e, a partir disso, utilizando o método de *MDP* para descrever as decisões de condução, o sistema criado recomenda as rotas em tempo real para aumentar a receita dos taxistas de carros elétricos. Ao analisar as contribuições desse projeto e ver os ganhos positivos, espera-se que os resultados dessa monografia possam ter também bons resultados, tendo em vista a similaridade em alguns aspectos e objetivos.

A abordagem utilizada por Kong et al. (2017) utiliza um modelo *Time-Location-Relationship (TLR)*, que tem a finalidade de identificar atributos como tempo, localização e relacionamento para poder realizar as previsões e poder recomendar Top-N áreas aos motoristas, ou seja, recomendar áreas rentáveis que estão adjacentes à localização atual do motorista. Esse é um trabalho que foca na redução do tempo de busca por passageiros, enquanto a monografia atual visa recomendar rotas com base em eficiência de combustível e na chance de encontrar passageiros.

Assim como Kong et al. (2017), um dos objetivos de Yuen et al. (2019) é justamente aumentar o número de clientes do operador de táxi e, consequentemente, maximizar os ganhos. Os autores utilizam da programação dinâmica para resolver o problema do caminho mais curto, onde ele procurará aqueles

caminhos mais curtos que tenham a maior quantidade de clientes. A diferença desse trabalho é que ele foca em táxis que fazem corridas compartilhadas.

Partindo da mesma ideia de minimizar os gastos dos taxistas e encontrar as melhores rotas para o encontro de passageiros, o diferencial da abordagem feita por [Wan, Ghazzai e Massoud \(2019\)](#) é que vai além dos serviços prestados por motoristas de táxis, englobando os táxis autônomos também. Nesse estudo, o algoritmo de Dijkstra foi utilizado para encontrar a rota mais rápida, dada todas as possibilidades, na finalidade de construir um sistema que oriente os motoristas de táxi para os “pontos quentes” e os táxis autônomos para os “pontos frios”, de maneira análoga, nesta monografia é utilizado o algoritmo de Dijkstra para encontrar a rota mais rápida mas sem englobar táxis autônomos.

Fazendo uma relação com esta monografia, a proposta trazida por [pa e Chakraborty \(2015a\)](#) é relevante para o escopo do projeto desta pesquisa, tendo em vista que para buscar uma rota ideal, é proposto um algoritmo de roteamento K-menor onde o custo do link vai variar de forma dinâmica e múltiplos destinos são igualmente aceitáveis, dessa forma podemos pensar em recomendar rotas para destinos distintos de uma mesma categoria, como exemplo, dadas as situações de trânsito, para qual posto de combustível direcionar o usuário.

[Isinkaye, Folajimi e Ojokoh \(2015\)](#) apresentam uma discussão em cima das técnicas mais tradicionais do campo da recomendação, a filtragem colaborativa e a filtragem baseada em conteúdo, além de explorar hibridizações feitas para aprimorar o desempenho dessas técnicas, elas podem ser utilizadas para personalizar ainda mais as recomendações. Ele traz também uma breve visão sobre as métricas de avaliação para os algoritmos apresentados, dividindo em métricas de precisão estatística e de suporte à decisão.

O estudo feito por [Raghuwanshi e Pateriya \(2019\)](#) trás várias técnicas de filtragem como as citadas na [Seção 2.1.1](#), ele traz pontos importantes também como as dificuldades dos sistemas com o problema de *Cold-Start*, que se dá pela escassez de dados para usuários e itens novos. Além disso, é feita uma análise de métodos de avaliações de desempenho dos [SR](#), o autor pontua métricas como [MAE](#), [RMSE](#), precisão, *recall* e *F-measure*.

[Pasinato et al. \(2013\)](#) apresenta uma metodologia para gerar dados sintéticos voltados à avaliação de [Sistemas de Recomendação Sensíveis ao Contexto \(CARS\)](#), com foco em recomendações de viagens. A proposta inclui um gerador de dados sintéticos que utiliza perfis de usuários e destinos modelados por variáveis aleatórias e suas [Sistemas de Recomendação Sensíveis ao Contexto \(CARS\)](#) para simular o comportamento de usuários em diferentes contextos. Além de permitir a avaliação de [CARS](#), o método visa compreender melhor como o contexto

influencia as avaliações dos usuários.

De maneira análoga a [Pasinato et al. \(2013\)](#) no estudo feito por [Silva, Silva e Porto \(2021\)](#) foi utilizado das [PDFs](#) e da técnica de histograma para modelar a [PDF](#). O foco desse trabalho foi criar uma ferramenta e uma nova técnica de aumento de dados para dados categóricos, onde a estratégia usada foi baseada em probabilidade sendo mais flexível do que as técnicas que utilizam *Machine Learning*. Essa ferramenta consegue gerar novas amostras de dados sintéticos treinada por uma base real.

3 Desenvolvimento

Neste capítulo, são apresentados os detalhes do desenvolvimento do sistema, incluindo as escolhas dos tipos de dados e as escolhas tecnológicas, os processos de implementação e a estrutura geral da solução proposta. Para uma melhor compreensão do processo. O desenvolvimento foi dividido nas seguintes etapas: (1) coleta e análise de dados, criando embasamento para a geração dos dados sintéticos, descrito na [Seção 3.1](#); (2) modelagem do problema, relatado na [Seção 3.2](#) as decisões para o desenvolvimento do projeto; (3) desenvolvimento do sistema, descrito na [Seção 3.3](#); e (4) experimentos, apresentados na [Capítulo 4](#). A [Figura 3.1](#) apresenta um fluxograma que fornece uma visão dos passos seguidos para o desenvolvimento do sistema.

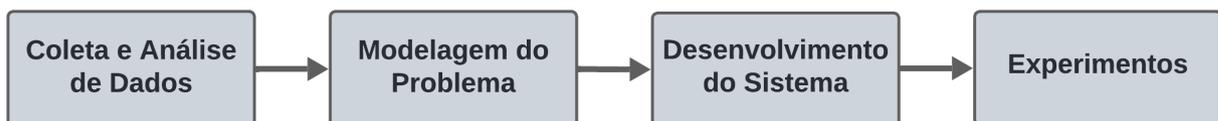


Figura 3.1 – Fluxograma do desenvolvimento do método base.

Este trabalho é uma pesquisa de natureza descritiva e exploratória, na qual foi orientado pelos requisitos levantados na fase de pesquisa e fundamentado pelas abordagens identificadas na revisão bibliográfica e tem como inspiração a abordagem utilizada por [Tu et al. \(2020\)](#).

3.1 Coleta e Análise de Dados

Indo contrário à abordagem feita por [Tu et al. \(2020\)](#), este atual estudo teve em vista trabalhar e se adaptar ao espaço e realidade do Brasil, por este motivo surgiu-se a primeira dificuldade: conseguir uma base de dados sólida e real que possuísse os dados [GPSs](#) dos veículos de transporte individual. Diante desse impasse, foi decidido utilizar uma base de dados da cidade de Nova Iorque e realizar os experimentos dentro do território nova iorquino, no distrito de Manhattan para ser mais preciso, que apresenta padrões de mobilidade urbana comparáveis a grandes cidades brasileiras, a exemplo da cidade de São Paulo.

Para fundamentar e auxiliar os processos seguintes, foi necessário:

- Entender os dados usados por [Tu et al. \(2020\)](#) e fazer adaptações considerando os veículos *flex* e o cenário que os viabilizasse;

- Mapear a cidade com os registros [GPS](#) e coletar análises estatísticas.

Sobre os processos em que exigia a utilização dos dados geográficos da cidade, foi utilizado o OpenStreetMap para extrair a rede viária de Manhattan. Com essas informações e os dados GPS dos táxis foi possível fazer a correspondência de mapas para construir as trajetórias espaço-temporais de cada táxi, ou seja, coletar com precisão todas as rotas percorridas pelos táxis fazendo a associação das ruas e rodovias com as posições coletadas pelas informações GPS. Além disso, por meio do [SUMO](#) e do histórico de corridas foi possível gerar novas corridas para os táxis levando em consideração as regiões com maior demanda de corridas.

Com as informações sobre as viagens de táxis e dados como localização, velocidade do veículo, quando e onde passageiros solicitam o serviço, tempo de cada viagem feita, além da localização e preço das recargas, foram importantes para este estudo e com essas informações foi possível fazer algumas análises a respeito da demanda dos táxis ao longo de um dia (24 horas) e durante a semana como podemos ver na [Figura 3.2](#) e na [Figura 3.3](#) respectivamente, além de conseguirmos fazer uma análise espacial das áreas da cidade que tiveram uma maior demanda, como visto na [Figura 3.4](#). Essas informações foram essenciais para ajudar no desenvolvimento dos próximos passos e na definição de alguns parâmetros.

A partir da análise da [Figura 3.2](#), foi possível identificar os horários com maiores picos de corridas em Nova Iorque. Com essas informações definiu-se a jornada de trabalho do taxista utilizada nos experimentos, compreendendo o intervalo de 12h às 22h. Já a [Figura 3.4](#) também foi importante para delimitação do escopo geográfico dos experimentos, como o distrito de Manhattan possui uma concentração maior de corridas, essa região foi escolhida como a área em Nova Iorque para aplicação dos experimentos.

Os dados de localização dos postos de combustíveis foram coletados com base no distrito de Manhattan para manter a conexão real com a rede viária da região. No entanto, para aproximar o estudo ao contexto brasileiro, para cada posto identificado, foram atribuídos valores para os preços do etanol e da gasolina de acordo com a média dos preços praticados na cidade de São Paulo durante os meses de fevereiro e março de 2025 ([ANP, 2025](#)). A partir desses valores médios, definiram-se limites superior e inferior, dentro dos quais os preços foram distribuídos aleatoriamente entre os postos simulados.



Figura 3.2 – Demanda de táxis durante o dia (24 horas).

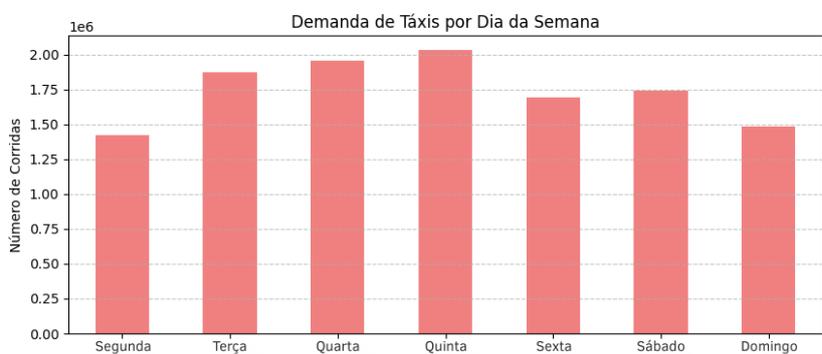


Figura 3.3 – Demanda de táxis por dias na semana.

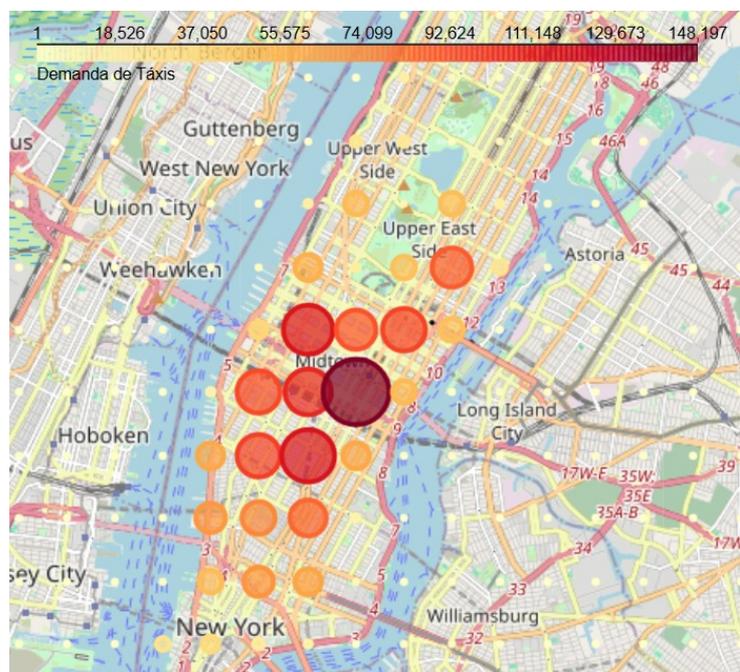


Figura 3.4 – Mapa das áreas de maior demanda.

3.2 Modelagem do Problema

Nesta subseção, descrevem-se as premissas adotadas, as variáveis consideradas e a formulação do problema no contexto da recomendação de rotas para táxis.

3.2.1 Estado do Táxi

É utilizado um **MDP** para descrever as decisões de direção e abastecimento do motorista. O estado de um táxi é definido como $s = (i, c, t) \in S$ em que $i \in I$ onde I é um conjunto de cruzamentos e i indicando a posição atual nesse cruzamento, c é a quantidade de combustível no tanque e $t \in T$ é o tempo atual em um dia, onde T é o conjunto de tempo representando 24h de um dia.

3.2.2 Ação do Táxi

A ação do veículo vai ser definida por $a = (i \rightarrow j)$, onde j é o próximo nó selecionado da rede. As ações poderão ser de cruzeiro ou de abastecimento.

As ações dos motoristas são definidas como $a = (i \rightarrow j) \in A$, em que $i \rightarrow j$ indica o deslocamento da posição i até j . Dois tipos de ações são definidos: cruzeiro e abastecimento, a ação de cruzeiro é representada por $a \in A_c$, que indica que o táxi está encontrando ou entregando algum passageiro. Sendo assim, para que o motorista desloque de i até j deve existir uma estrada r_{ij} que conecte i a j . A ação de abastecimento é representada por $a \in A_r$, que indica que o motorista irá até um posto em j e abastecerá o tanque.

3.2.3 Receita Líquida Esperada

Em se tratando da **RLE**, um motorista aumentará esse montante ao realizar ações, em uma corrida de i para k , a receita líquida se dá por

$$RLE(i, k) = R^*(i, k) - C(i, k)$$

em que $R^*(i, k)$ indica a receita gerada pela viagem de i para k que pode ser calculada levando em conta o tempo e a distância total e $C(i, k)$ é o custo do combustível caso seja feito um abastecimento. É importante lembrar que para encontrar a rota ideal devemos encontrar a sequência de ações que maximize a receita líquida do motorista. O cálculo do custo do combustível é feito pela seguinte fórmula:

$$C(i, k) = distancia * \frac{P}{Autonomia}$$

Onde:

- P é o preço do combustível (álcool ou gasolina).
- *Autonomia* representa a autonomia do veículo que pode variar a depender do combustível que está sendo utilizado.

3.2.4 Modelo dos Táxis

Os táxis são representados como entradas em um dicionário, onde cada táxi (ex.: taxi_0, taxi_1) possui atributos como destino atual, próximo destino, combustível atual, e lucro total. Inicializados em um ponto fixo (195743214#0), os táxis operam independentemente por 36000 segundos, alternando entre estados de cruzeiro e abastecimento. A lógica de decisão prioriza as rotas viáveis mais curtas e postos mais baratos ou mais pertos.

Os atributos de cada táxi incluem:

- **Destino Atual:** Rua atual (ex.: -630828563#2).
- **Próximo Destino:** Rua do destino ou “None” se ocioso.
- **Ocupado:** Booleano indicando se o táxi está em uma corrida ou não.
- **Corridas Completas:** Contador de corridas concluídas.
- **Distância Total:** Distância percorrida em metros.
- **Tempo Total:** Tempo acumulado das corridas em segundos.
- **Combustível Atual:** Nível de combustível em litros.
- **Lucro Total:** Receita das corridas, calculada como $\text{distancia_km} * 4.25$
- **Custo de Combustível:** Custo acumulado de reabastecimentos.

3.3 Desenvolvimento do Sistema

A rede viária, definida em *manhattan.net.xml*, modela Manhattan como um grafo direcionado, com nós (interseções) e arestas (ruas) com IDs únicos. Carregada pela biblioteca *sumolib*, ela permite converter coordenadas geográficas em edges e calcular distâncias, além disso, a rede fornece informações como o comprimento das arestas (em metros) e os tipos de vias. A rede é responsável por alguns papéis principais:

- **Mapeamento Geográfico:** Utiliza a técnica de *map matching* para converter as coordenadas de latitude e longitude em arestas. Ela permite uma fácil construção das trajetórias espaço-temporais, ou seja, coletar com precisão todas as rotas percorridas pelos táxis fazendo a associação das ruas e rodovias com as posições coletadas pelas informações **GPS**.
- **Cálculo de Rotas e Distâncias:** A função *traci.simulation.findRoute* utiliza o algoritmo de Dijkstra para selecionar caminhos otimizados se baseando no comprimento das arestas. Além disso, utiliza-se o comprimento das arestas para fazer os cálculos de gasto de combustível e do lucro de cada corrida.

3.3.1 Histórico de Corridas

O arquivo *trips_monday.csv* contém dados reais de corridas de táxi em Manhattan do dia 7 de março de 2016. As colunas principais incluem a data/hora do início da corrida, as coordenadas do local de partida e as coordenadas do destino final. Com a função *carregar_historico*, as corridas são organizadas em uma grade de 20x20 regiões, com base nas coordenadas do ponto inicial da corrida. Cada região (ex.: 10_12) é associada a um peso de demanda, calculado pela função *calcular_pesos_demanda*, que reflete a frequência de corridas em cada área.

O peso de uma região r é calculado por:

$$\text{peso}_r = \frac{\text{Número de corridas na região } r}{\text{Total de corridas}}$$

Esse peso é utilizado para: (1) Busca Probabilística de Passageiros, no momento em que o táxi está ocioso em uma aresta a probabilidade de encontrar um passageiro é igual ao peso da região; (2) Patrulhamento Otimizado, que escolhe arestas vizinhas com maior peso, maximizando a chance de encontrar passageiros.

O pseudocódigo da busca probabilística é:

```
# Busca probabilística de passageiros
se taxi.ocioso e sem_proximo_destino:
    aresta_atual = taxi.destino_atual
    regioao = mapear_aresta_para_regiao(aresta_atual)
    peso = pesos_demanda[regiao] ou 0.01
    se random.uniform(0, 1) < peso:
        corrida = escolher_destino(historico, regioao)
        novo_origem = aresta_atual
        novo_destino = converte_coord_para_aresta(corrida.destino_latitude,
        corrida.destino_longitude)
        se novo_destino e novo_origem != novo_destino:
```

```

rota, consumo = calcular_rota_viavel(novo_origem,
novo_destino, combustivel)
se rota:
    taxi.proximo_destino = novo_destino
    taxi.ocupado = Verdadeiro
    atualizar_combustivel(taxi, consumo)

```

A busca probabilística verifica se um táxi que está ocioso encontra um passageiro com base no peso de demanda da região em que ele se encontra, calculado como $\text{Peso}_r = (\text{Corridas na região}) / (\text{Total de corridas})$. A aresta atual é mapeada para uma célula, se o motorista encontrar passageiros, uma corrida do histórico *trips_monday.csv* é escolhida, seu destino convertido em aresta, e a rota é calculada com *traci.simulation.findRoute*. Se viável, o táxi atualiza destino, estado (ocupado) e combustível. Caso contrário, passa ao patrulhamento e *escolher_patrolha* seleciona a aresta vizinha com maior peso:

```

# Escolha de patrulha otimizada
funcao escolher_patrolha(aresta_atual, combustivel, autonomia, taxi_id):
    vizinhos = obter_vizinhos_viaveis(aresta_atual)
    rotas_viaveis = []
    para vizinho em vizinhos:
        rota, consumo = calcular_rota_viavel(aresta_atual, vizinho,
        combustivel)
        se rota:
            regioao = mapear_aresta_para_regiao(vizinho)
            peso = pesos_demanda[regiao] ou 0.0
            rotas_viaveis.append((vizinho, consumo, peso))
    se rotas_viaveis:
        escolha = max(rotas_viaveis, key=lambda x: x[2])
        retornar escolha[0], escolha[1]
    retornar None, 0

```

A função *escolher_patrolha* move o táxi para a aresta vizinha que possui maior peso de demanda quando nenhum passageiro é encontrado. Então obtém vizinhos viáveis, calcula as rotas e consumos, mapeia cada vizinho para uma célula e seleciona o com maior peso. Retorna a aresta e consumo ou None se não houver opções, sinalizando possível reabastecimento.

3.3.2 Postos de Combustível

Os postos de combustível são definidos no arquivo *postos_manhattan.csv*, que lista 17 postos com seus respectivos nós na rede (ex.: 42440138) e preços por litro de gasolina e álcool. A função *carregar_postos* mapeia esses nós para arestas acessíveis na rede, utilizando um raio de busca de 500 metros.

3.3.3 Funções de Decisão

A Figura 3.5 apresenta um fluxograma que representa a lógica de decisão utilizada pelos táxis durante a simulação. Inicialmente, o táxi parte de um ponto inicial verificando a quantidade de combustível no tanque. Caso o nível de combustível esteja abaixo do limite estabelecido, o taxista segue para abastecer num posto de combustível mais próximo ou mais barato. Caso contrário, ele começa a sua busca por passageiros considerando sempre os caminhos com a maior probabilidade de encontrar passageiros. Ao encontrar um passageiro, ele realiza a viagem até o destino final, se não ele continua procurando por novos passageiros. Esse processo se repete até o fim da simulação. O sistema implementa quatro funções principais para coordenar o comportamento dos táxis:

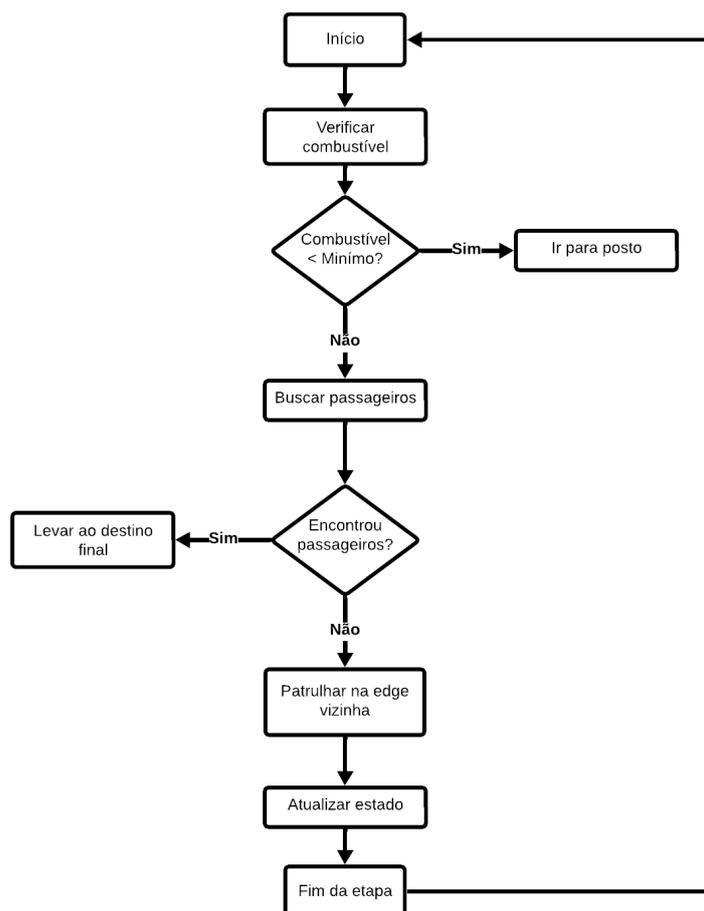


Figura 3.5 – Fluxograma da lógica de decisão dos táxis.

- **Cálculo de Rotas (*calcular_rota_viavel*):** Calcula rotas viáveis entre origem e destino, usando a função *traci.simulation.findRoute* da biblioteca *TraCI* em

dois modos (DEFAULT e COMBINED). Essa função também filtra as rotas com consumo de combustível menor ou igual ao disponível e escolhe a rota mais curta.

- **Escolha de Postos (*escolher_posto*):** Seleciona o posto com menor preço ou o mais perto entre os acessíveis, otimizando custos.
- **Escolha de Patrulha (*escolher_patrulha*):** Escolhe uma aresta vizinha acessível para patrulha, se baseando na probabilidade de encontrar passageiros.
- **Garantia de Rota Válida (*garantir_rota_valida*):** Corrige rotas inválidas, atribuindo uma nova rota a um vizinho viável, evitando paradas indesejadas.

4 Resultados

Este capítulo apresenta os resultados obtidos a partir dos experimentos realizados seguindo a metodologia e desenvolvimento descritos anteriormente. Os dados provenientes dos experimentos foram analisados e interpretados com o intuito de avaliar o sistema proposto neste trabalho, identificar padrões e validar a eficácia das implementações. Para melhor entendimento, este capítulo foi dividido em duas seções: (1) os parâmetros definidos para a simulação foram explicados na [Seção 4.1](#); (2) as análises feitas a partir dos experimentos foram exploradas na [Seção 4.2](#).

4.1 Parâmetros

Os seguintes parâmetros são importantes para a simulação:

Parâmetro	Valor
Duração	36000 segundos (10 horas)
Número de táxis	10
Autonomia	18.5 km/L e 12.8 km/L
Taxa por km	R\$4.25
Combustível mínimo	10.0 L
Combustível inicial	50.0 L
Ponto inicial	195743214#0

Tabela 4.1 – Parâmetros da simulação.

Para a definição de certos parâmetros do experimento, foi necessário realizar pesquisas complementares. O veículo considerado nas simulações foi o Toyota Corolla, pois esse apresenta o melhor consumo médio, com de 12,8 km/l com etanol e 18,5 km/l com gasolina ([ADOP, 2024](#)). O período da simulação foi definido como uma segunda-feira, com a jornada de trabalho do taxista ocorrendo entre 12h e 22h, totalizando 10 horas de atividade. Essa escolha está de acordo com a Lei nº 13.103/2015, que estabelece uma carga horária padrão de até 8 horas diárias para motoristas profissionais, podendo ser estendida em até 2 horas extras, respeitando o limite legal de 10 horas de trabalho por dia.

A quantidade de táxis na simulação foi de 10 para cada *baseline* para garantir uma viabilidade computacional das simulações no [SUMO](#). A medida que o número de táxis na simulação vai aumentando, um problema surgia. Quando haviam muitos veículos transitando na rede viária e um carro se aproximava da

borda do mapa ele era removido da simulação. Por conta dessa dificuldade, o número de veículos utilizados na simulação foi de 10.

4.2 Experimentos

Para avaliar o sistema implementado cinco cenários foram avaliados e descritos em sequência. As principais métricas avaliadas são a **RLE**, número de corridas, distância percorrida e custo do combustível, essas métricas foram escolhidas por estarem alinhadas com o objetivo do trabalho, maximizar a receita dos motoristas considerando a probabilidade de encontrar passageiros e o custo dos reabastecimentos.

- **RANDOM**: Táxis patrulham aleatoriamente, sem considerar pesos de demanda ou otimização de reabastecimento, simulando uma estratégia não otimizada.
- **MAXPROB1**: Utiliza busca probabilística de passageiros, recomenda a rota mais rápida e seleciona o posto de gasolina mais barato.
- **MAXPROB2**: Similar ao MAXPROB1, mas seleciona o posto mais barato para álcool.
- **MAXPROB3**: Similar ao MAXPROB1, mas seleciona o posto de gasolina mais próximo.
- **MAXPROB4**: Similar ao MAXPROB3, mas seleciona o posto mais próximo para álcool.

As Tabelas: [Tabela 4.2](#), [Tabela 4.3](#) e [Tabela 4.4](#) apresentam as médias das métricas para os cenários com 100%, 80% e 60% de combustível inicial, respectivamente. Esses cenários foram pensados para aqueles casos em que nem sempre um motorista começa sua jornada de trabalho com 100% de combustível no tanque.

Baseline	Comb.	Corr.	Dist. (km)	Lucro (R\$)	Custo (R\$)	RLE (R\$)	C. Final (L)
RANDOM	Álcool	39.50	112.58	479.37	0.00	479.37	20.86
RANDOM	Gasolina	38.60	113.70	483.23	0.00	483.23	29.54
MAXPROB1	Gasolina	39.20	117.03	501.64	0.00	501.64	29.67
MAXPROB2	Álcool	38.80	110.49	469.56	0.00	469.56	21.44
MAXPROB3	Gasolina	40.20	113.14	480.88	0.00	480.88	29.91
MAXPROB4	Álcool	39.70	122.60	520.89	0.00	520.89	19.79

Tabela 4.2 – Médias das métricas para 10 táxis por baseline (100% de combustível).

Com a [Tabela 4.2](#) e a [Figura 4.1](#) se vê que com 100% de combustível, MAXPROB4 (álcool, posto mais próximo) obteve o maior RLE (R\$520.89), superando MAXPROB1 (gasolina, posto mais barato, R\$501.64) e RANDOM (gasolina, R\$483.23). A estratégia de reabastecimento no posto mais próximo maximizou o tempo para corridas, resultando em maior distância percorrida (122.60 km) e lucro (R\$520.89), além de que os dois *baselines* que usam a estratégia de abastecer em postos próximos foram o que tiveram uma maior quantidade na média de corridas (40.20 e 39.70).

Baseline	Comb.	Corr.	Dist. (km)	Lucro (R\$)	Custo (R\$)	RLE (R\$)	C. Final (L)
RANDOM	Álcool	39.20	114.29	485.30	50.54	434.76	20.71
RANDOM	Gasolina	39.60	113.81	484.02	0.00	484.02	19.45
MAXPROB1	Gasolina	39.80	113.91	484.12	0.00	484.12	20.02
MAXPROB2	Álcool	36.30	117.87	497.93	50.88	447.05	21.02
MAXPROB3	Gasolina	38.20	110.76	470.80	0.00	470.80	20.30
MAXPROB4	Álcool	38.10	118.14	496.87	34.17	462.70	17.66

Tabela 4.3 – Médias das métricas para 10 táxis por baseline (80% de combustível).

Analisando os resultados na [Tabela 4.2](#) e a [Figura 4.1](#), referente ao cenário de 80% de combustível, MAXPROB1 (gasolina, R\$484.12) liderou, pode-se concluir que esse ganho se deve a maior autonomia da gasolina (18.5 km/L) no veículo *flex*. Em cenários com 100% e 80% do combustível inicial, pode-se observar um custo de abastecimento nulo para certas *baselines*, isso mostra que o combustível nos dois experimentos foi suficientes para cobrir toda a jornada de trabalho do motorista, tendo em vista as autonomies para cada tipo de combustível utilizado.

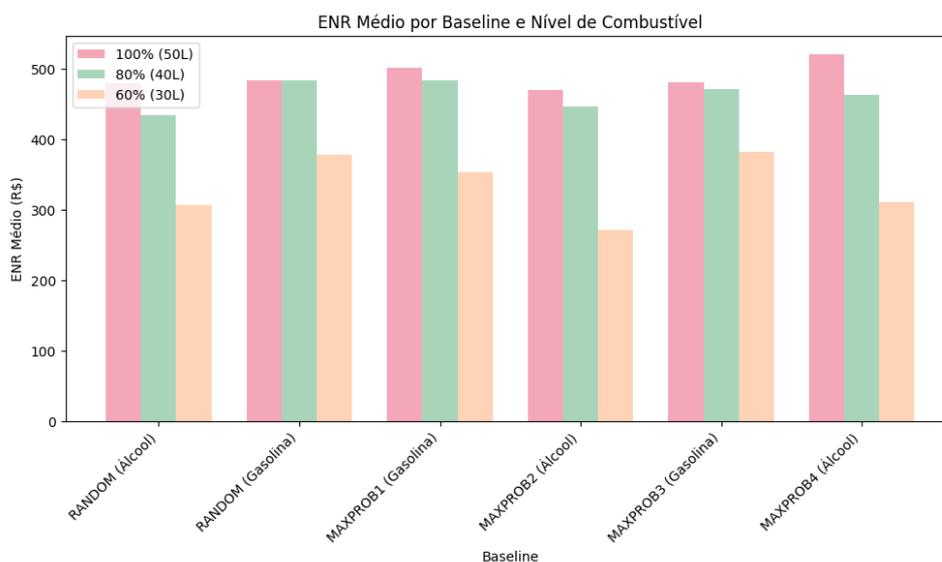


Figura 4.1 – RLE médio por baseline e nível de combustível.

Baseline	Comb.	Corr.	Dist. (km)	Lucro (R\$)	Custo (R\$)	RLE (R\$)	C. Final (L)
RANDOM	Álcool	38.20	112.19	476.83	169.58	307.25	41.65
RANDOM	Gasolina	38.10	112.55	478.37	100.77	377.60	24.35
MAXPROB1	Gasolina	38.90	112.76	479.23	125.39	353.84	29.30
MAXPROB2	Álcool	37.60	103.60	441.98	170.69	271.29	42.83
MAXPROB3	Gasolina	39.20	119.61	508.37	125.75	382.62	28.65
MAXPROB4	Álcool	36.00	113.53	482.52	171.54	310.98	41.62

Tabela 4.4 – Médias das métricas para 10 táxis por baseline (60% de combustível).

E já na [Tabela 4.2](#) para o experimento com 60% de combustível, MAXPROB3 (gasolina, R\$382.62) teve o maior RLE, indicando que a escolha do posto mais próximo com gasolina reduz reabastecimentos em cenários onde o combustível está limitado.

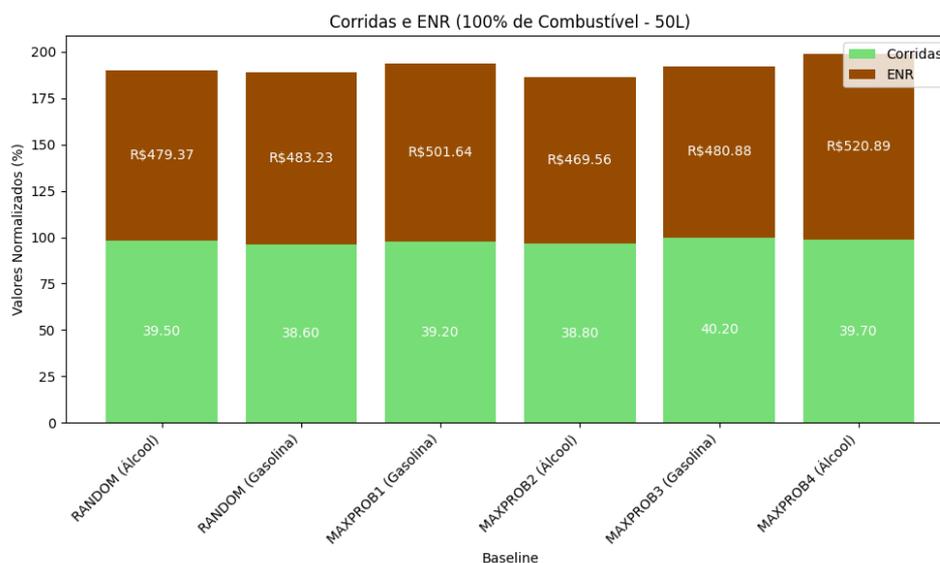


Figura 4.2 – Número de corridas e RLE (100% de combustível).

A [Figura 4.2](#) compara o número de corridas e o RLE com 100% de combustível. MAXPROB3 (gasolina, 40.20 corridas) se destacou em número de corridas, mas MAXPROB4 (álcool, R\$520.89) tem o maior RLE, devido a corridas mais longas (122.60 km). A [Figura 4.3](#) mostra que, com 60% de combustível, MAXPROB3 (gasolina, 39.20 corridas, R\$382.62) teve destaque em RLE, enquanto baselines com álcool têm desempenho inferior devido a necessidade de reabastecimentos frequentes por conta da sua autonomia ser baixa.

Um último experimento feito foi pra simular motoristas que tem uma tendência a abastecer mesmo o que o tanque não esteja vazio, para isso consideramos uma variável *alfa* que representa o limite percentual de combustível no tanque que gera a decisão de reabastecimento. Esse experimento foi pensado para si-

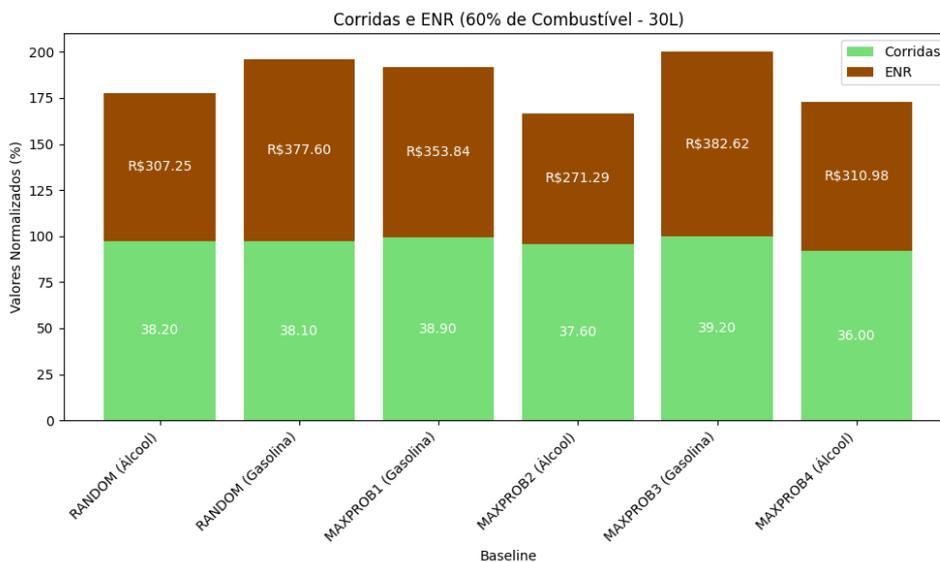


Figura 4.3 – Número de corridas e RLE (60% de combustível).

mular casos onde há aqueles motoristas que abastecem quando o combustível atinge a reserva, por exemplo, e aqueles que não esperam o nível de combustível diminuir muito. Foram feitos experimentos com os valores de 20% (10L) e 50% (25L), representando cenários de motoristas cautelosos que abastecem cedo e aqueles menos cautelosos que esperam o tanque estar mais vazio. A Tabela 4.5 apresenta as médias do RLE para os experimentos com a variável *alfa* sendo 20% (10L) e 50% (25L).

Na Figura 4.4 o RLE com *alfa* sendo 20% (10L). MAXPROB4 (álcool, R\$518.21) tem o maior ENR, devido a corridas mais longas (média de 121.57 km). A Figura 4.5 indicam que, com *alfa* sendo 50% (25L) MAXPROB3 (gasolina, R\$514.14) em RLE. Pode-se perceber também que as *baselines* com álcool têm RLE inferior devido a baixa autonomia que o veículo *flex* tem quando está consumindo álcool.

Baseline	Combustível	RLE (20%)	RLE (50%)
RANDOM	Álcool	R\$440.13	R\$369.09
RANDOM	Gasolina	R\$489.94	R\$494.98
MAXPROB1	Gasolina	R\$497.43	R\$511.61
MAXPROB2	Álcool	R\$469.56	R\$365.29
MAXPROB3	Gasolina	R\$473.94	R\$514.14
MAXPROB4	Álcool	R\$518.21	R\$400.13

Tabela 4.5 – Médias de RLE para limites de combustível de 20% e 50%.

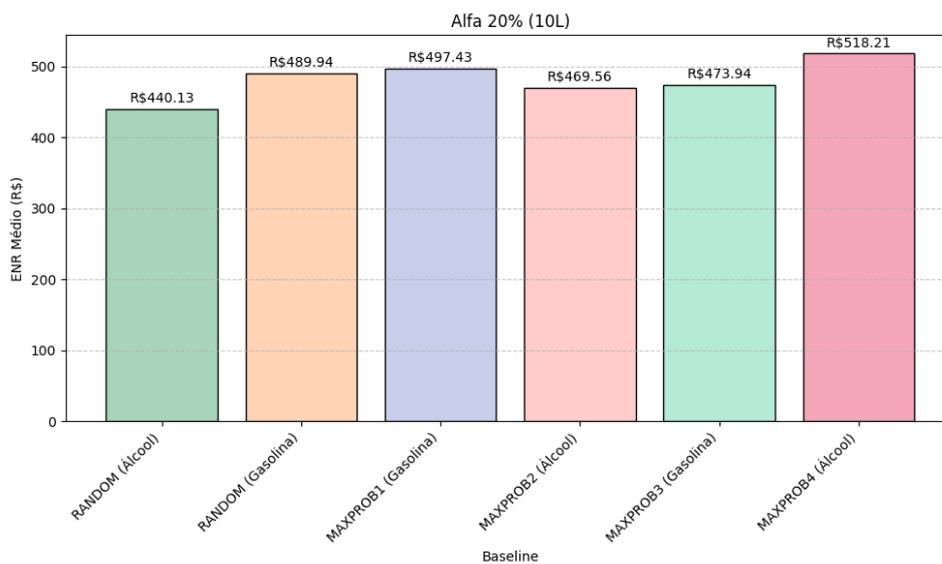


Figura 4.4 – RLE médio com 20% de combustível (10L).

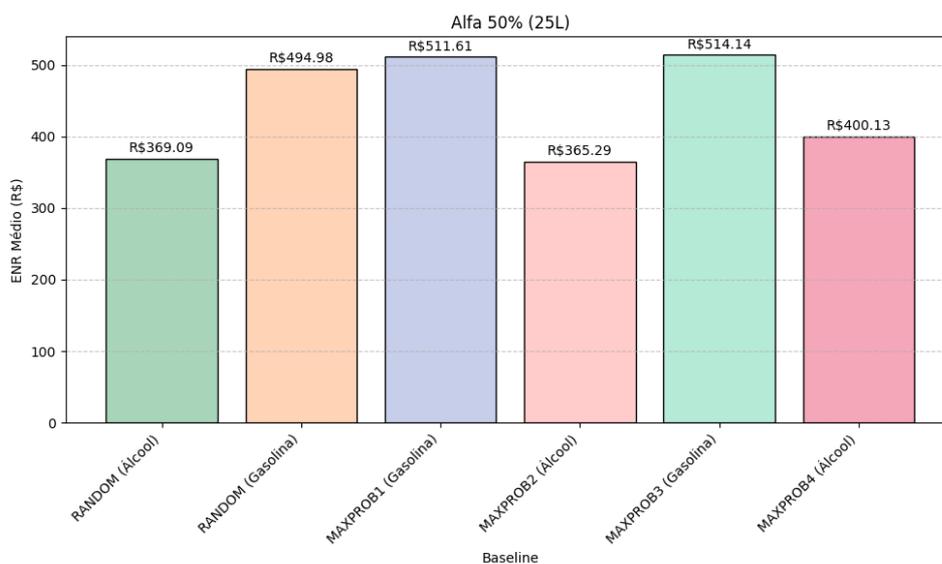


Figura 4.5 – ENR médio com 50% de combustível (25L).

5 Considerações Finais

Este trabalho desenvolveu um sistema de recomendação de rotas para taxistas, utilizando busca probabilística e patrulhamento otimizado, adaptado para veículos *flex* no contexto brasileiro. Foram testadas várias estratégias de abastecimento e escolha de combustível, denominadas MAXPROB, que combinam o tipo de combustível utilizado (álcool ou gasolina) com a estratégia de escolha do posto (mais próximo ou mais barato).

Os experimentos evidenciaram que a estratégia MAXPROB4 (que utiliza álcool e abastece no posto mais próximo) obteve o maior valor de [RLE](#) quando os veículos partiram com 100% do tanque cheio. Já a MAXPROB1 (que utiliza gasolina e abastece no posto mais barato) apresentou melhor desempenho com 80% de combustível, enquanto a MAXPROB3 (gasolina e posto mais próximo) foi mais eficaz com 60%. Observou-se que a gasolina foi mais eficiente em cenários com níveis mais baixos de combustível, pois exigiu menos reabastecimentos. Por outro lado, o uso do álcool, que possui menor autonomia, resultou em uma maior frequência de paradas para abastecimento, impactando negativamente a [RLE](#).

A principal contribuição deste estudo é a implementação de uma abordagem não personalizada com busca probabilística, que modela a probabilidade de encontrar passageiros com base em pesos de demanda a partir de registros de outras corridas de táxis adaptada para o contexto brasileiro por meio de veículos *flex*. Com escolha dinâmica entre álcool e gasolina e estratégias de reabastecimento, percebe-se a vantagem de postos próximos e bem localizados e a eficiência da gasolina diante de recursos limitados.

Durante o processo de desenvolvimento deste trabalho, foram identificadas algumas limitações. Uma delas foi a grande dificuldade em encontrar dados reais de táxis brasileiros, sendo necessário utilizar a base de dados de Nova Iorque e adaptá-la ao contexto brasileiro. Além disso, durante o desenvolvimento com o [SUMO](#) muitas dificuldades foram encontradas devido à complexidade da ferramenta, que exigiu um grande tempo de aprendizado, um dos problemas enfrentados com o [SUMO](#) foi a remoção automática de táxis ao se aproximarem ou ultrapassarem a borda da rede viária. Por fim, o número de táxis por *baseline* foi de certo modo um limitante para os experimentos, uma vez que uma simulação com uma quantidade muito maior de veículos demandaria um tempo de processamento muito grande.

5.1 Trabalhos Futuros

Para trabalhos futuros, almeja-se a adição de novas variáveis no sistema, como preferências individuais dos motoristas, algumas condições de tráfego e custos extras como pedágios, com o intuito de aumentar a robustez e realismo do sistema. Além disso, busca-se viabilizar a obtenção de dados reais de táxis no Brasil por meio de possíveis parcerias com empresas de táxi ou de aplicativos de mobilidade para que sejam realizadas simulações que reflitam com ainda mais realismo o contexto brasileiro. Outra ideia é tornar o sistema personalizado, considerando os padrões e costumes de cada motorista individualmente. Por fim, planeja-se também realizar simulações com uma quantidade maior de veículos e para isso recursos mais eficientes serão buscados. Todas essas melhorias têm o intuito de ampliar e aumentar ainda mais os ganhos deste trabalho.

Referências

- ADOP. *União Nacional da Bioenergia*. 2024. <https://www.udop.com.br/noticia/2024/06/28/ate-18-5-km-l-inmetro-traz-novo-ranking-dos-carros-flex-mais-econ-ocirc-micos.html>. Acessado em: 16 abril. 2025.
- ANFAVEA. *Anuário Anfavea Indústria Automobilística Brasileira*. 2024. <https://anfavea.com.br/site/anuarios-2/>. Acessado em: 24 set. 2024.
- ANP. *Agência Nacional do Petróleo, Gás Natural e Biocombustíveis*. 2025. <https://www.gov.br/anp/pt-br/assuntos/precos-e-defesa-da-concorrenca/precos/arq-sintese-semanal/2025/fevereiro/sintese-precos-n07.pdf>. Acessado em: 16 abril. 2025.
- AZAMBUJA, R. X. d.; MORAIS, A. J.; FILIPE, V. Teoria e prática em sistemas de recomendação. *RCC - Revista de Ciências da Computação*, v. 16, p. 24, 2021. Disponível em: <<https://journals.uab.pt/index.php/rcc/article/view/264>>.
- BEHRISCH, M.; BIEKER-WALZ, L.; ERDMANN, J.; KRAJZEWICZ, D. Sumo – simulation of urban mobility: An overview. In: . [S.l.: s.n.], 2011. v. 2011. ISBN 978-1-61208-169-4.
- CAZELLA, S. C.; NUNES, M.; REATEGUI, E. A ciência da opinião: Estado da arte em sistemas de recomendação. *André Ponce de Leon F. de Carvalho; Tomasz Kowaltowski.(Org.). Jornada de Atualização de Informática-JAI*, p. 161–216, 2010.
- HERLOCKER, J.; TERVEEN, L.; LUI, J. C.; RIEDL, T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, v. 22, p. 5–53, 01 2004.
- IBM. *União Nacional da Bioenergia*. 2024. <https://www.ibm.com/br-pt/think/topics/data-augmentation>. Acessado em: 16 abril. 2025.
- ISINKAYE, F.; FOLAJIMI, Y.; OJOKOH, B. Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, v. 16, n. 3, p. 261–273, 2015. ISSN 1110-8665. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1110866515000341>>.
- JORDON, J.; SZPRUCH, L.; HOUSSIAU, F.; BOTTARELLI, M.; CHERUBIN, G.; MAPLE, C.; COHEN, S.; WELLER, A. Synthetic data – what, why and how? 05 2022.
- KONG, X.; XIA, F.; WANG, J.; RAHIM, A.; DAS, S. Time-location-relationship combined service recommendation based on taxi trajectory data. *IEEE Transactions on Industrial Informatics*, PP, p. 1–1, 03 2017.
- MARAN, V.; MARUYAMA, M.; SILVEIRA, L. Aplicação de técnicas de recomendação de recursos educacionais em um campus universitário. *Ciência e Natura*, v. 45, p. e17, 10 2023.

- PA, C. Sub-r; CHAKRABORTY, G. Dynamic distribute route recommendation system for multiple destinations. In: *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. [S.l.: s.n.], 2015. p. 1–5.
- PA, C. Sub-r; CHAKRABORTY, G. Dynamic distribute route recommendation system for multiple destinations. In: *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. [S.l.: s.n.], 2015. p. 1–5.
- PASINATO, M.; MELLO, C. E.; AUFAURE, M.-A.; ZIMBRÃO, G. Generating synthetic data for context-aware recommender systems. In: *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. [S.l.: s.n.], 2013. p. 563–567.
- QU, M.; ZHU, H.; LIU, J.; LIU, G.; XIONG, H. A cost-effective recommender system for taxi drivers. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014. Disponível em: <<https://api.semanticscholar.org/CorpusID:4228349>>.
- RAGHUWANSHI, S. K.; PATERIYA, R. K. Recommendation systems: Techniques, challenges, application, and evaluation. In: BANSAL, J. C.; DAS, K. N.; NAGAR, A.; DEEP, K.; OJHA, A. K. (Ed.). *Soft Computing for Problem Solving*. Singapore: Springer Singapore, 2019. p. 151–164. ISBN 978-981-13-1595-4.
- SILVA, H.; SILVA, R.; PORTO, F. Sagad: Synthetic data generator for tabular datasets. In: *Anais do XXXVI Simpósio Brasileiro de Bancos de Dados*. Porto Alegre, RS, Brasil: SBC, 2021. p. 1–12. ISSN 2763-8979. Disponível em: <<https://sol.sbc.org.br/index.php/sbbd/article/view/17861>>.
- TANDON, R.; VERMA, A.; GUPTA, P. Rvtn: Recommender system for vehicle routing in transportation network. In: . [S.l.: s.n.], 2022. p. 352–356.
- TU, W.; KE, M.; ZHANG, Y.; XU, Y.; HUANG, J.; DENG, M.; LI, Q. Real-time route recommendations for e-taxis leveraging gps trajectories. *IEEE Transactions on Industrial Informatics*, PP, 04 2020.
- UIT. *Measuring the Information Society Report*. 2015. <https://www.itu.int/en/ITU-D/Statistics/Pages/publications/mis2015.aspx>. Acessado em: 21 set. 2024.
- UIT. *Measuring digital development: Facts and Figures 2023*. 2023. <https://www.itu.int/itu-d/reports/statistics/facts-figures-2023/>. Acessado em: 21 set. 2024.
- WAN, X.; GHAZZAI, H.; MASSOUD, Y. Online recommendation system for autonomous and human-driven ride-hailing taxi services. In: . [S.l.: s.n.], 2019. p. 351–354.
- WANG, R.; CHOW, C.-Y.; LYU, Y.; LEE, V. C. S.; KWONG, S.; LI, Y.; ZENG, J. Taxirec: Recommending road clusters to taxi drivers using ranking-based extreme learning machines. *IEEE Transactions on Knowledge and Data Engineering*, v. 30, n. 3, p. 585–598, 2018.

YUEN, C.; SINGH, A.; GOYAL, S.; RANU, S.; BAGCHI, A. Beyond shortest paths: Route recommendations for ride-sharing. In: . [S.l.: s.n.], 2019. p. 2258–2269. ISBN 978-1-4503-6674-8.