



Otimização de Rotas para Coleta de Dados Imobiliários

Aplicação de Teoria dos Grafos e Algoritmos de Roteamento

Amon Lemes dos Santos	2023005290
Lucas Salles Granado	2023006878
Sammuel Gonçalves Reis	2023004480
Ulisses Douglas de Paula Assis Junior	2024001849

Universidade Federal de Itajubá (UNIFEI)

Curso de Sistemas de Informação

Disciplina: Grafos (SMAC03)

Equipe e contribuições

Amon Lemes dos Santos

Exploração de alternativas e apoio no desenvolvimento técnico.

Sammuel Gonçalves Reis

Integração do pipeline e ajustes no dataset final.

Lucas Salles Granado

Implementação das rotas e otimização para múltiplos agentes.

Ulisses Douglas de Paula Assis Junior

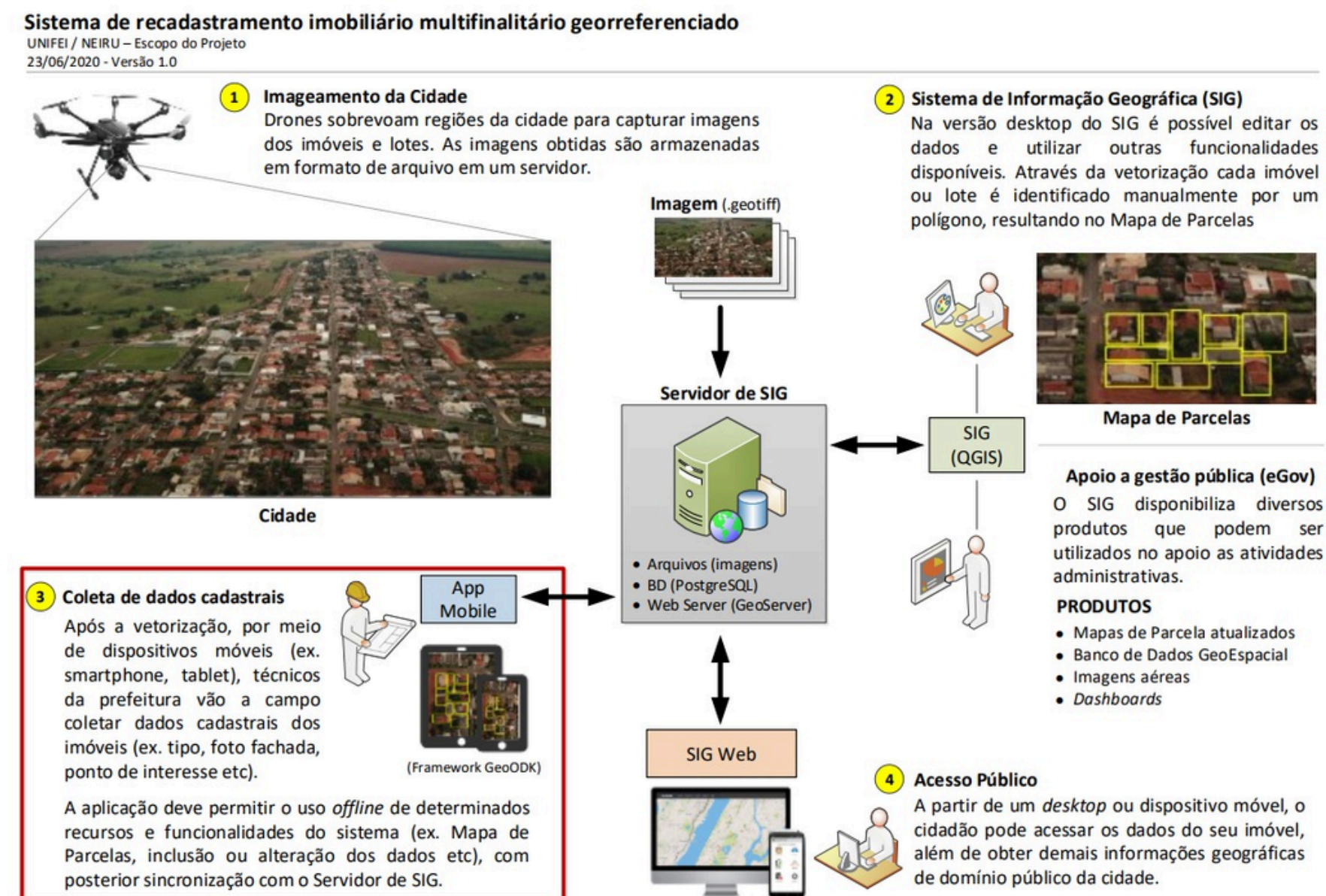
Organização do projeto, automação da execução e apoio no desenvolvimento.

Contexto Geral do Recadastramento Imobiliário

Um sistema completo de recadastramento imobiliário envolve diversas etapas, desde o imageamento da cidade até o acesso público às informações

Essas etapas incluem captura de imagens via drone, vetorização no SIG, coleta em campo e posterior sincronização com o banco de dados georreferenciado.

Nosso trabalho foca em otimizar o deslocamento dos agentes que realizam a coleta presencial, reduzindo tempo e custo operacional.



Visão em alto nível do processo de recadastramento imobiliário Multifinalitário Georreferenciado.

O recadastramento imobiliário exige visitas presenciais a todos os lotes, gerando um alto custo logístico

A coleta de dados imobiliários é fundamental para a justiça fiscal e planejamento urbano municipal. Cada lote deve ser visitado presencialmente para validar informações cadastrais.

Em cidades como **Elói Mendes/MG**, este processo envolve:

Desafios Logísticos

Centenas de lotes

Dispersos em toda a cidade, exigindo rotas eficientes para minimizar tempo e custo de deslocamento.

Custo Operacional

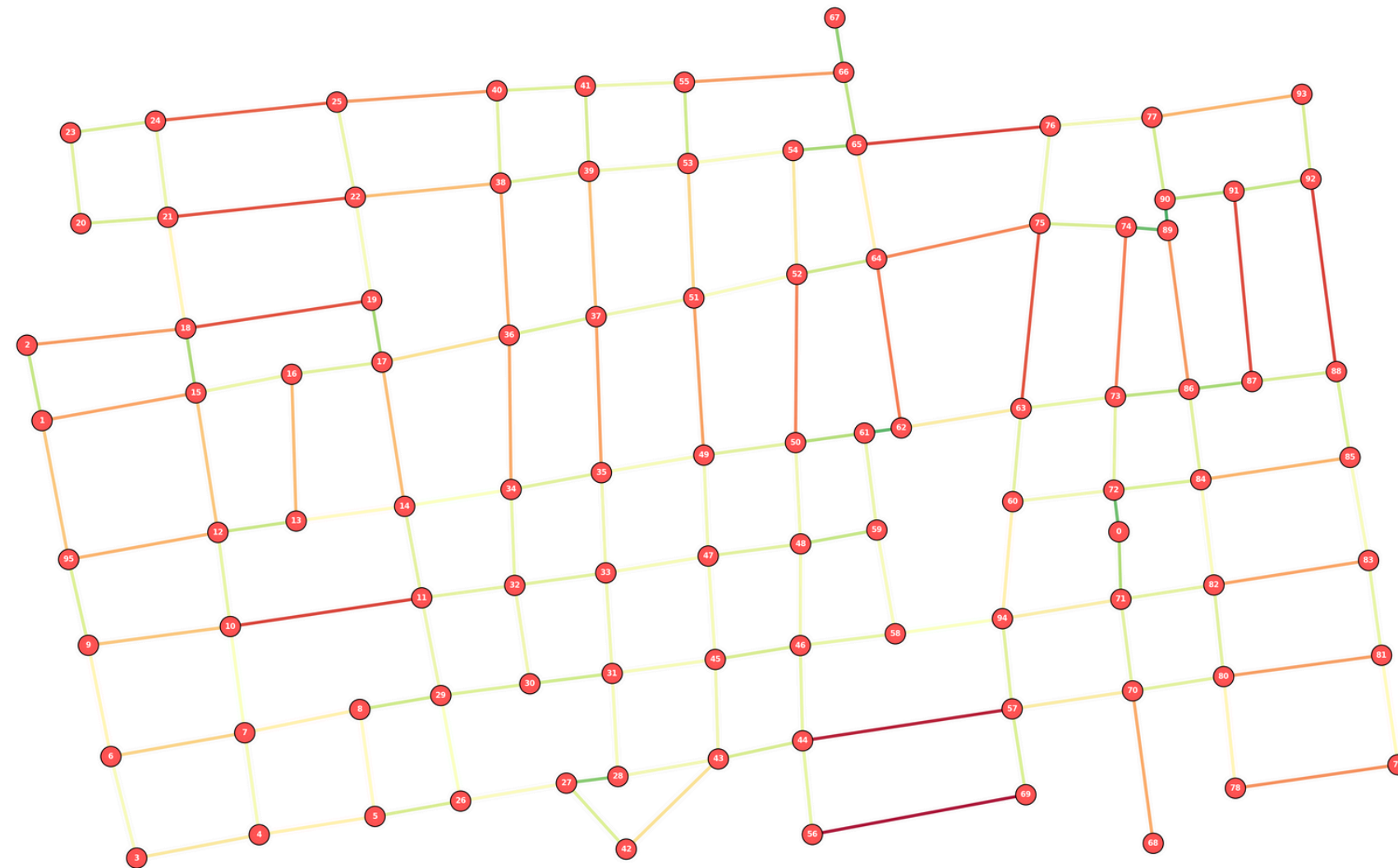
Alto

Tempo dos agentes, desgaste e custo operacional são significativos.



Ortofoto de Elói Mendes. Único dataset fornecido para modelagem do problema

**O mapa da cidade deve ser transformado em um modelo matemático para permitir
o roteamento eficiente**



Conversão de rede viária em grafo: nós (cruzamentos) e arestas (ruas)

O algoritmo de Edmonds-Johnson garante que todas as ruas sejam visitadas com o menor custo total possível

O **Problema do Carteiro Chinês (CPP)** é resolvido através do algoritmo de Edmonds-Johnson, que garante um caminho que percorre todas as arestas do grafo pelo menos uma vez.

Passo 1

Identificar todos os vértices com grau ímpar no grafo de ruas.

Passo 2

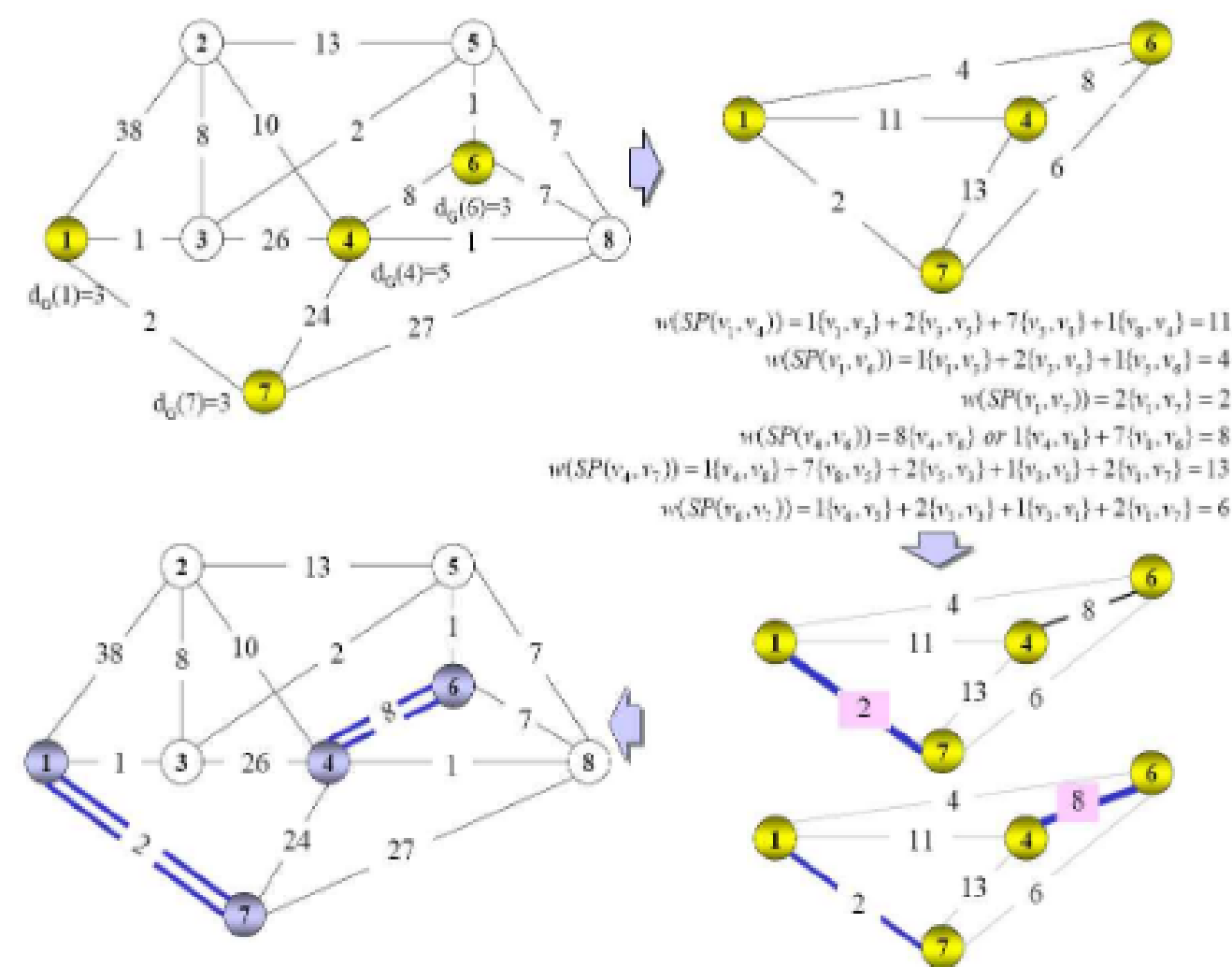
Calcular o emparelhamento mínimo entre esses vértices ímpares.

Passo 3

Duplicar as arestas do emparelhamento no grafo original.

Resultado

Grafo resultante possui todos os vértices com grau par, permitindo um caminho Euleriano que percorre todas as ruas.



Estratégia de Pós-Processamento e Divisão de Rotas

Transformar uma rota única contínua (Solução CPP) em rotas viáveis para múltiplos agentes, garantindo conectividade com a base.

Passo 1

Calcula o custo total e define uma meta de corte para dividir o trabalho igualmente entre N agentes.

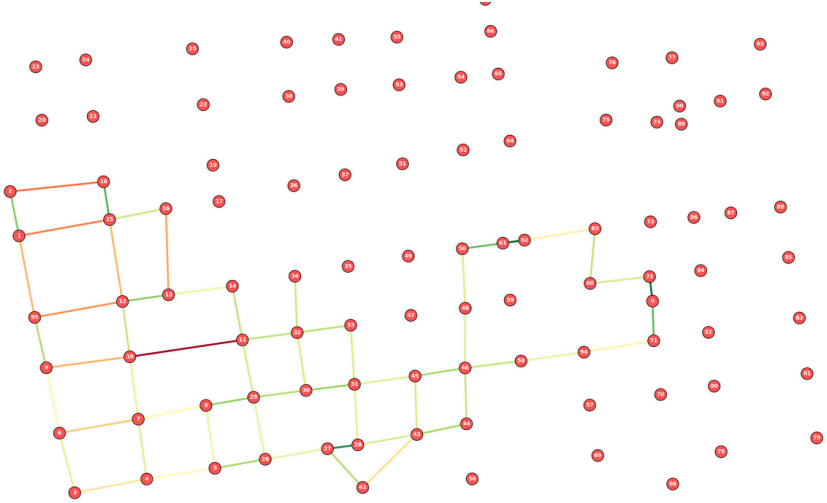
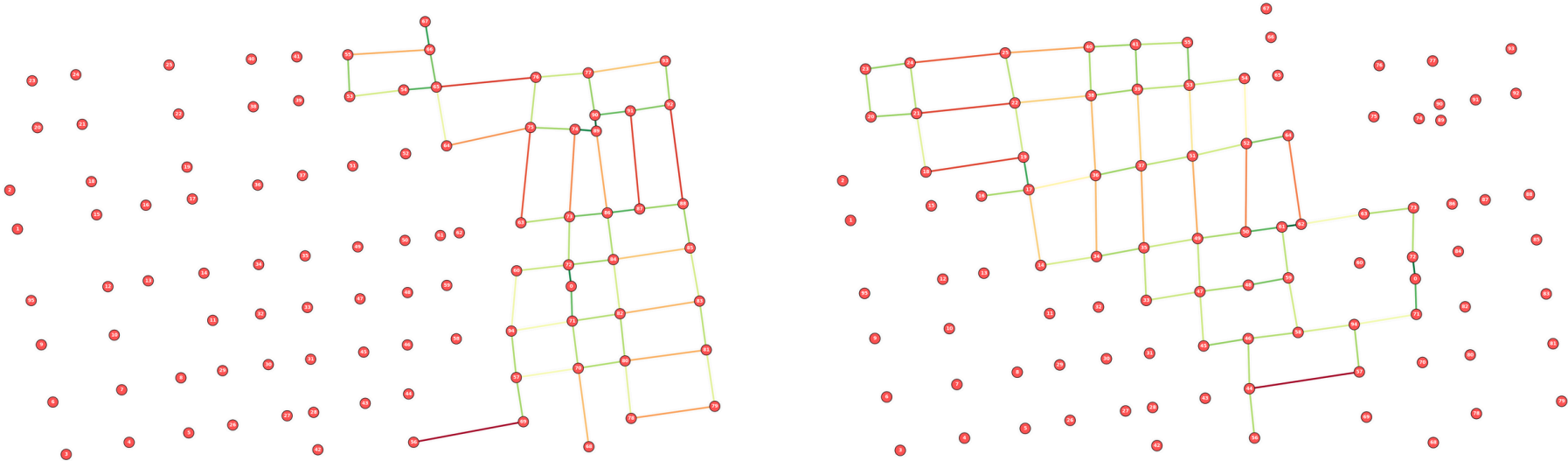
Passo 2

Utiliza Dijkstra para calcular o caminho da Base → Início do Setor de cada agente.

Passo 3

Utiliza Dijkstra para calcular o retorno do Fim do Setor → Base.

Resultado individual para cada agente, contendo apenas o seu subgrafo de operação + deslocamentos



Resultado do particionamento para 3 agentes

A complexidade da coleta de dados é capturada ao ponderar as arestas pela densidade de casas, não apenas pela distância

Uma rota não é custosa apenas pelo seu **comprimento**. O tempo que o agente gasta depende também do **número de casas** a visitar em cada segmento de rua.

$$P = \frac{d}{1,4} + 20n$$

Comparação: Mesma Distância, Esforço Diferente



Poucas casas: peso baixo

Muitas casas: **peso alto**



Nosso modelo incorpora **ambos os fatores**:
distância + densidade de casas = custo real da aresta.

- P é o **Peso** total (o resultado final).
- d é a **distância** (`distancia_m`).
- n é o **número de casas** (`numero_de_casas`).
- 1,4 é a velocidade de caminhada (1.4 m/s).
- 20 é o tempo de serviço por casa (20 s).

Ferramentas visuais permitem a validação da coerência e da eficiência das rotas geradas

Python

Linguagem de programação para processamento de dados e algoritmos de otimização

OSMnx

Extração de dados geoespaciais do OpenStreetMap e construção de grafos de ruas

Folium

Visualização interativa de mapas com rotas sobrepostas em imagens de satélite



Coerência: Todas as ruas são visitadas sem desvios desnecessários



Eficiência: Rotas minimizam distância e tempo total de coleta

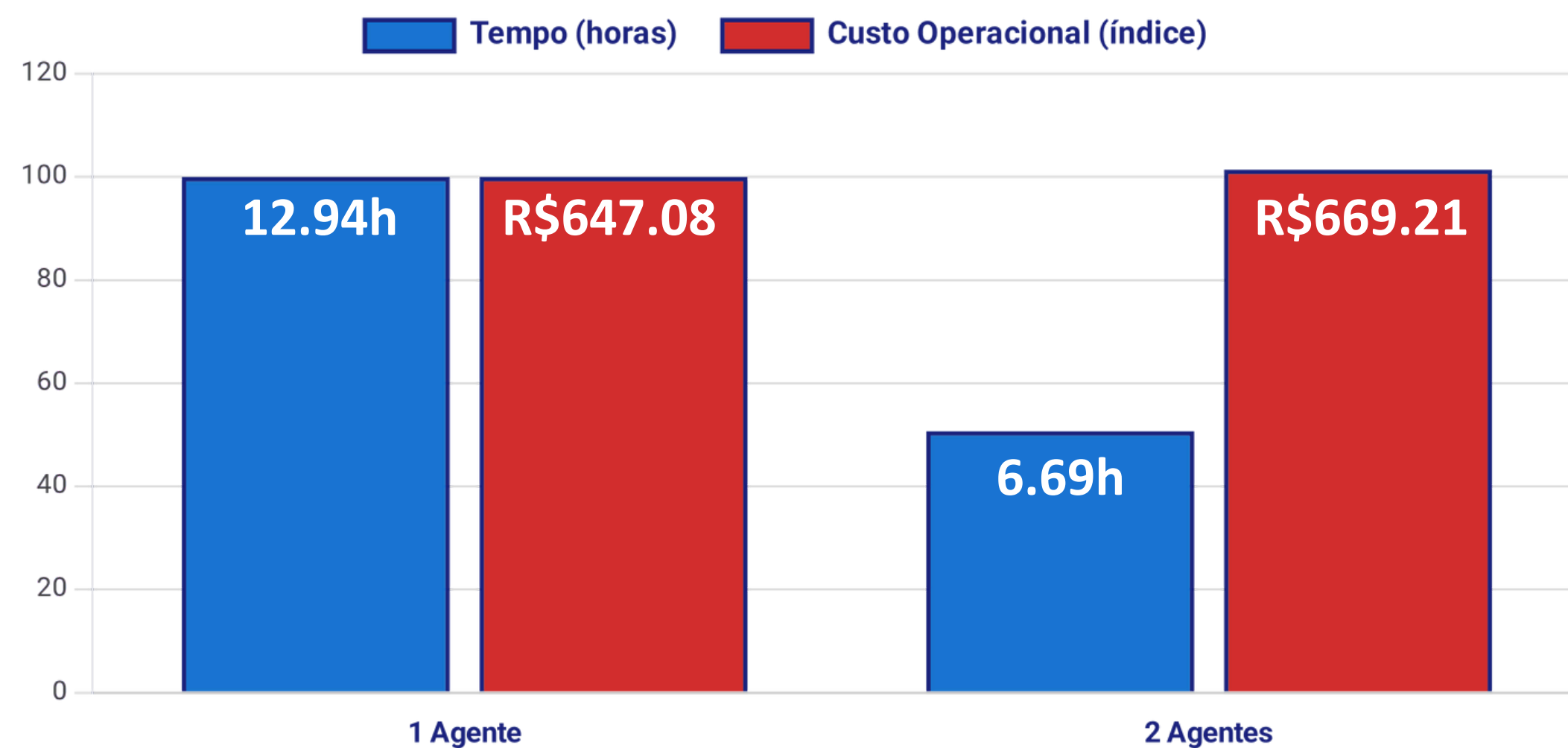


Praticidade: Agentes de campo podem verificar e validar as rotas no mapa



Viabilidade: Solução é escalável para outras cidades e contextos?

A configuração com dois agentes reduziu o tempo de coleta em 49,3% com um aumento de custo operacional de apenas 1,4%



-49,3%

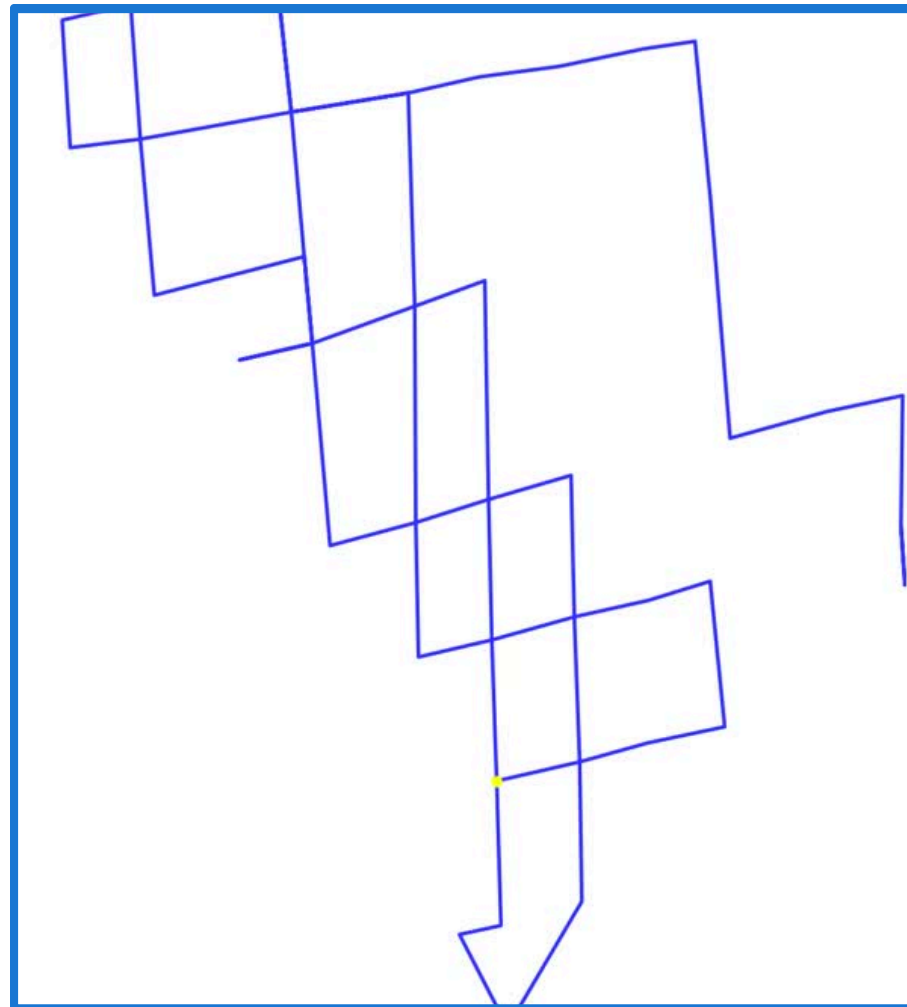
Redução de Tempo

+1,4%

Aumento de Custo

A setorização eficiente das rotas demonstra visualmente o equilíbrio da carga de trabalho entre os agentes

Agente 1 - Rota Azul

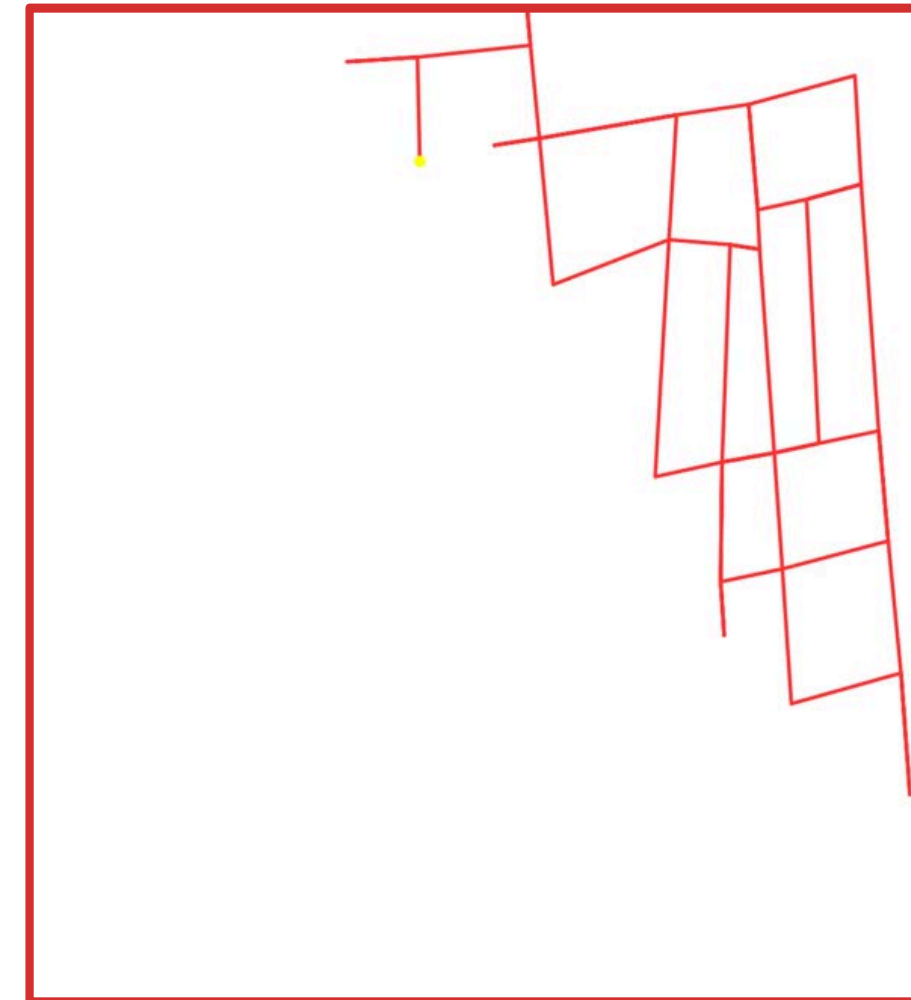


Carga de Trabalho

49.3%

Tempo: ~6.5 horas

Agente 2 - Rota Vermelha



Carga de Trabalho

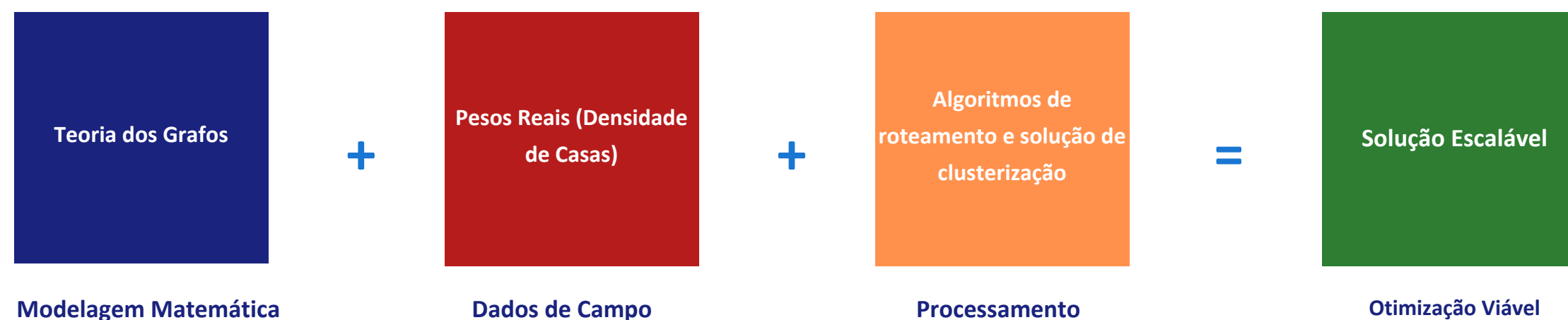
50.7%

Tempo: ~6.69 horas

Resultado: Equilíbrio Perfeito

As duas rotas otimizadas apresentam uma distribuição **praticamente equilibrada** da carga de trabalho, com diferença de apenas **1,4%** no custo operacional total.

A abordagem híbrida de Grafos e Pesos Reais validou uma solução escalável e economicamente viável para a prefeitura



Benefícios Alcançados



Redução de Tempo

49,3% de economia no tempo total de coleta com dois agentes



Escalabilidade

Metodologia aplicável a outras cidades e contextos de roteamento



Custo Marginal

Aumento de apenas 1,4% no custo operacional, altamente aceitável



Viabilidade Econômica

Solução tecnicamente robusta e economicamente justificável para prefeituras