



UM ALGORITMO PARA GERAÇÃO DE ROTAS PARA COLETA DE RESÍDUOS SÓLIDOS NA CIDADE DE MANAUS UTILIZANDO DADOS IOT
AN ALGORITHM FOR THE GENERATION OF ROUTES FOR THE COLLECTION OF SOLID WASTE IN THE CITY OF MANAUS USING IOT DATA

LOPES, DIEGO; NEVES, NAYRA; SOARES, ANDRÉ; BASTOS, MOISES; VALENZUELA, WALTER

UNIVERSIDADE DO ESTADO DO AMAZONAS

diml.eng@uea.edu.br, ngn.eng16@uea.edu.br, alsoares@uea.edu.br, mpbastos@uea.edu.br, wvalenzuela@uea.edu.br

Resumo – A Internet das Coisas (IoT) atualmente vem sendo amplamente estudada na área de cidades inteligentes para sensoriamento, localização, rastreamento e gerência de coisas inteligentes. A geração de rotas para caminhões coletores de resíduos sólidos da cidade de Manaus pode ser inserida em uma aplicação de Iot. O problema de roteamento de veículos trata de um conjunto de rotas de veículos de custo mínimo, originando e terminando em um depósito central e possui heurísticas de desenvolvimento. Este trabalho propõe um algoritmo para geração de múltiplas rotas, utilizando a heurística orientada ao tempo e ao vizinho mais próximo, com uma origem e um destino especificados para o recolhimento de resíduos nos pontos coletáveis com volumes diversos, em seguida apresenta os resultados de simulações quantitativas feitas a partir dos parâmetros propostos, além das rotas a distância percorrida e o tempo necessário para cada trajeto.

Palavras-chave: Iot. VRP. Vizinho mais próximo.

Abstract - The Internet of Things (IoT) is currently being widely studied in the area of smart cities for sensing, locating, tracking and managing smartthings. The generation of routes for trucks collecting solid waste from the city of Manaus can be inserted in an Iot application. The vehicle routing problem deals with a set of minimum cost vehicle routes, originating and ending in a central warehouse and has development heuristics. This work proposes an algorithm for the generation of multiple routes, using the time-oriented heuristic and the closest neighbor, with a specified origin and destination for the collection of waste at the collectable points with different volumes, then presents the results of simulations. measurements made from the proposed parameters, in addition to the routes the distance traveled and the time required for each route.

Keywords: Iot. VRP. Nearest neighbor.

I. INTRODUÇÃO

A Internet das coisas, do inglês *The Internet Of Things* (IoT) é um dos termos mais utilizados quando se fala na atual fase da industrialização. A IoT é o modo como todas as coisas estão conectadas à internet e se comunicando entre si, através de dispositivos de sensoriamento inteligentes e softwares que transmitem dados para uma rede. Ela alcança o objetivo de identificar, localizar, rastrear, monitorar e gerenciar coisas inteligentes, (Solomon, 1987).

A internet das coisas nos possibilita integrar praticamente tudo que está ao nosso redor. As informações compartilhadas permitem que sejam feitas inúmeras

aplicações de inovações que variam das mais diversas áreas da sociedade, como meio ambiente, indústria, comércio e *smart cities* e tem por objetivo melhorar a qualidade de vida das pessoas.

O planejamento de roteirização de caminhões coletores de lixo da cidade de Manaus é uma das aplicações na qual a IoT pode otimizar com auxílio de sistemas de sensoriamento inteligente. Em geral, as gerações de itinerários precisam diminuir os custos e maximizar a produtividade. Por esse motivo existem soluções relacionadas ao Problema de Roteamento de Veículos, do inglês Vehicle Routing Problem (VRP), que diz respeito a um problema de otimização combinatória, que envolve o projeto de um conjunto de rotas de veículos de custo mínimo, originando e terminando em um depósito central, para uma frota de veículos que atende um conjunto de clientes com demandas conhecidas.

O VRP diz respeito ao serviço de logística. As mercadorias de n clientes possuem uma demanda de mercadorias e estas são entregues por uma frota de veículos homogêneos. Os veículos precisam realizar um percurso saindo do local do depósito e entregando as mercadorias para um subconjunto de clientes, levando em consideração a demanda e o retorno ao local inicial. As rotas de cada veículo não podem ultrapassar a capacidade do veículo e o tempo máximo para finalizar a rota. Todos os requisitos e restrições operacionais dos clientes precisam ser atendidos e o custo de transporte global minimizado. Esse custo pode ser monetário, de distância, de tempo ou de outra forma. (Arakaki, 1998).

Formalmente, o VRP é definido em um grafo direcionado $G = (V, A)$ com o conjunto de vértices V e o conjunto de arcos A . O conjunto de vértices V consiste no conjunto de clientes $N = \{1, \dots, n\}$ e os vértices 0 e $n + 1$ que representam o depósito no início e no final. Cada cliente $i \in N$ está associado a uma janela de tempo $[a_i, b_i]$, um tempo de serviço não negativo s_i , um intervalo de demanda $[q_{min} i, q_{max} i]$ e um lucro $e_i \geq 0$ por unidade de demanda entregue. Além disso, uma janela de tempo $[a_0, b_0] = [a_{n+1}, b_{n+1}]$ é associada aos depósitos para modelar o horizonte de planejamento (Tilk et al, 2019).

Dentre os vários tipos de algoritmos de roteamento, Solomon (1987) aborda em seu estudo sobre algoritmos para roteamento de veículos e problemas de agendamento com restrições de janela de tempo. Nele os algoritmos são

divididos em métodos sequenciais e paralelos. Onde procedimentos sequenciais constroem uma rota de cada vez até que todos os clientes sejam agendados. E os paralelos são caracterizados pela construção de uma série de rotas simultaneamente.

Esse trabalho, de caráter exploratório e de abordagem quali-quantitativa, descreve o desenvolvimento de um algoritmo para geração de múltiplas rotas para coleta de resíduos sólidos na cidade de Manaus utilizando dados de um sistema IoT, com base nas heurísticas orientada para o tempo e do vizinho mais próximo abordado por Solomon (1987), onde todas as rotas são iniciadas encontrando-se o cliente não roteado "mais próximo" ao depósito.

A métrica utilizada nesta abordagem tenta explicar a proximidade geográfica e temporal dos clientes. Deixe o último cliente na rota parcial atual ser cliente i , e deixe j denotar qualquer cliente não roteado que possa ser visitado em seguida. Em seguida, a métrica utilizada, c_{ij} , mede a distância direta entre os dois clientes, d_{ij} , a diferença de tempo entre a conclusão do serviço em i e o início do serviço na j , T_{ij} , e a urgência de entrega ao cliente j , v_{ij} , como expressa pelo tempo restante até o último possível início de serviço do veículo (Solomon, 1987).

II. REFERENCIAL TEÓRICO

2.1 – Iot (Internet of Things)

Iot é uma grande rede de equipamentos (coisas) ligados entre si capazes de gerar uma grande quantidade de dados que podem ser utilizados em diversas aplicações. Em seu artigo, Talari et al (2017), afirma que a IoT é capacitada pela expansão de várias coisas e equipamentos de comunicação e envolvem equipamentos inteligentes, como, celulares, eletrodomésticos ou sensores que colaboram para um objetivo comum.

2.2 – Vehicle Routing Problem (VRP)

Segundo Barrier (2003), o problema de roteamento de veículo consiste em vários clientes, cada um exigindo que um peso específico das mercadorias seja entregue. Os veículos despachados de um único depósito devem entregar as mercadorias necessárias e retornar ao depósito. Cada veículo pode ter um peso limitado e também pode ser restringido na distância total que pode percorrer. Apenas um veículo pode visitar cada cliente. O problema é definir um conjunto de rotas de entrega que atenda a esses requisitos e ofereça um custo total mínimo. Na prática, isso geralmente é considerado equivalente a minimizar a distância total percorrida ou a minimizar o número de veículos utilizados e, em seguida, minimizar a distância total para esse número de veículos. Uma das abordagens para solucionar este problema utiliza a heurística do vizinho mais próximo.

2.3 – Vizinho mais Próximo

O algoritmo de vizinho mais próximo é eficaz na solução de problemas altamente complexos, como o VRP. Estes são principalmente problemas de otimização. A técnica do vizinho mais próximo K geralmente usando a distância euclidiana ou a comparabilidade cosseno entre as tuplas (Mohammed et al, 2017). Na heurística do vizinho mais próximo é adicionado, a cada iteração, o vértice mais próximo do último já inserido. Quando todos os vértices são visitados, o procedimento é terminado.

III. METODOLOGIA

O estudo proposto, de caráter descritivo e exploratório, com abordagem quantitativa e qualitativa da implementação de um algoritmo de geração de rotas para caminhões coletores de resíduos sólidos dentro da cidade de Manaus. Para a sua implementação foi utilizada uma heurística orientada a tempo e vizinho mais próximo, apresentada por Solomon(1987).

Cada ponto coletável dentro da cidade foi considerado como um nó de um grafo. Por necessidades de execução das atividades de coleta, cada ponto coletável (ou nó do grafo) foi associado a uma região geográfica. Para cada região é gerado um grafo contendo seus pontos coletáveis.

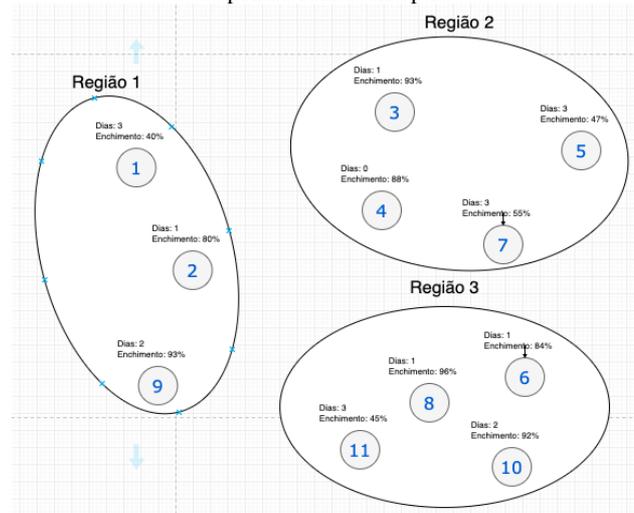
Os pontos coletáveis possuem um peso que é calculado a partir dos dados obtidos pelo sensoriamento e armazenado em servidores na nuvem e é definido por uma equação condicional que leva em conta o seu nível de enchimento, a quantidade de dias que não é coletado e o nível de gás presente em seu interior. Para a execução do algoritmo só serão considerados os pontos que satisfizeram a condição da equação (1) a seguir.

$$(Nível > 80\%) \text{ ou} \tag{1}$$

$$(Nível > 20\% \text{ e Dias} > 2) = \text{verdadeiro}$$

Nesta etapa, o resultado são grafos, por regiões geográficas, totalmente desconectados, possuindo apenas pontos coletáveis aptos, como apresentado na figura 1.

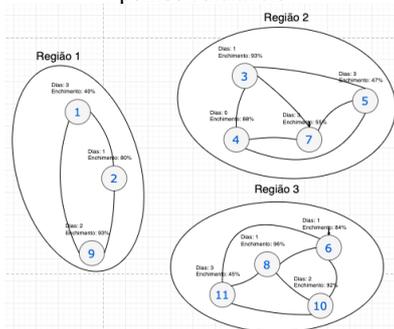
Figura 1 – Exemplo de grafo desconectado por região contendo os pontos coletáveis aptos



Fonte: Elaboração dos autores.

Para que o algoritmo possa ser executado, é necessário um grafo não direcionado e totalmente conectado. Para isso, a cada nó foram criadas arestas para todos os demais nós do grafo da região, como mostrado na figura 2. Tais arestas possuem pesos que são calculados de acordo com a distância em metros e o tempo em segundos entre cada ponto.

Figura 2 – Exemplo de grafos por região com arestas entre todos os pontos coletáveis



Fonte: Elaboração dos autores.

Como entrada do algoritmo, estão: origem, destino, tempo máximo de coleta para cada caminhão, uma lista de caminhões contendo sua capacidade e uma lista de grafos por região não direcionados e totalmente conectados, contendo os pontos de coleta (nós do grafo) e suas arestas encontradas na etapa anterior. Apresentamos a seguir um pseudocódigo do algoritmo gerado:

geradorRotas(origem, destino, Tmax, Capacidade, caminhões, grafos)
noAtual = origem
listaRotas = [[noAtual]]

para i em grafos faça:

enquanto grafo[i].vertices.tamanho > 0 faça:

rota = listaRotas.tamanho

se Tmax > listaRotas[rota].tempo e Capacidade >

listaRotas[rota].capacidade:

listaRotas[rota].add(vizinhoMaisProximo(noAtual, grafo[i]))

As etapas de preparação para que se possa entrar com os dados necessários para o gerador de rotas, incluem:

- I. Análise dos pontos coletáveis aptos a serem coletados,
- II. Divisão dos pontos coletáveis em regiões geográficas da cidade de Manaus (norte, sul, leste, oeste, centro-oeste e centro-sul), para isso foi utilizado a *Application Programming Interface* (API) do Google Maps,
- III. geração do grafo, onde seus nós são os pontos coletáveis e as arestas são as distância (calculada pela API do Google) entre cada nó e os demais pontos.

IV. RESULTADOS

Os resultados abaixo foram obtidos por meio de simulação de dados, onde foram gerados aleatoriamente os dados dos pontos de coleta (capacidade, nível de enchimento e dias sem coleta).

3.1 – Resultados da Simulação

Foram utilizados dados de pontos coletáveis (coordenadas no mapa de Manaus, nível de enchimento e dias sem coleta) gerados aleatoriamente e um computador MacBook Air 2005, com memória RAM de 4 gigabytes, 1600 megahertz, DDR3 e um processador Intel Core I5 Dual-Core, de 1,6 gigahertz. Os pontos coletáveis utilizados para a simulação são apresentados na tabela 1.

Tabela 1 – Pontos coletáveis utilizados para a simulação.

Nr o	Latitude/Longitude	Região	Enchimento (l)	Dias sem Coleta
1	-3.000140 / -60.002441	zona-norte	150	1
2	-3.004274 / -59.980428	zona-norte	180	1
3	-3.000118 / -59.972010	zona-norte	120	2
4	-3.013564 / -59.939042	zona-norte	200	1
5	-3.014602 / -59.938829	zona-norte	180	1
6	-3.031978 / -59.989389	zona-norte	140	2
7	-3.032043 / -59.985927	zona-norte	100	2
8	-3.031977 / -59.989465	zona-norte	160	1
9	-3.035027 / -59.939318	zona-leste	192	1
10	-3.044733 / -59.944959	zona-norte	100	1
11	-3.050639 / -59.944833	zona-leste	70	2
12	-3.062960 / -59.950434	zona-leste	150	2
13	-3.075378 / -59.955009	zona-leste	180	1
14	-3.075872 / -59.957190	zona-leste	160	1
15	-3.058301 / -60.006438	zona-centro-sul	170	1
16	-3.066278 / -59.990212	zona-centro-sul	140	2
17	-3.081527 / -60.024595	zona-centro-sul	160	1
18	-3.091780 / -60.017621	zona-centro-sul	160	1
19	-3.093316 / -60.021974	zona-centro-sul	160	1
20	-3.091220 / -60.017978	zona-oeste	120	1
21	-3.094113 / -60.009411	zona-centro-sul	100	2
22	-3.104551 / -60.026484	zona-centro-sul	120	2
23	-3.104443 / -60.012300	zona-centro-sul	160	2
24	-3.101366 / -60.056590	zona-oeste	140	2
25	-3.101133 / -60.053886	zona-oeste	160	2
26	-3.107280 / -60.053046	zona-oeste	160	1
27	-3.125645 / -60.026459	zona-centro-sul	160	1
28	-3.129877 / -60.023451	zona-sul	180	1
29	-3.127342 / -60.006121	zona-sul	120	2
30	-3.128988 / -59.997934	zona-sul	90	2

Os pontos coletáveis, descritos na tabela 1, foram inseridos utilizando latitude e longitude de suas coordenadas geográficas e foram separados em regiões, sendo estas

denominadas: norte, leste, centro-sul, oeste e sul dentro da cidade de Manaus. A figura 3 mostra os pontos dispostos no mapa.

Figura 3 – Pontos coletáveis exibidos no mapa de Manaus



Fonte: Elaboração dos autores.

O algoritmo foi implementado utilizando linguagem JavaScript por meio do framework Node.js em sua versão 10.16.3.

A tabela 2 mostra quais pontos coletáveis pertencem a cada rota, as distâncias necessárias para completar a rota em metros e o tempo total em minutos de cada trajeto com base nas características dos pontos coletáveis da tabela 1.

Tabela 2 – Rotas geradas pelo algoritmo.

Rota	Pontos Coletáveis (Nro)	Distância Total (m)	Tempo Total Aproximado (min)
1	3, 7, 6	34000	59
2	8, 4, 5, 10, 2	52200	87
3	11, 12, 14	26000	53
4	13, 9	29000	54
5	16, 21, 23	22900	41
6	17, 19, 22, 18	16600	38
7	27, 15	25700	51
8	26, 25, 24, 20	18600	39
9	29, 30, 28	17600	47

A distância total e o tempo total aproximado mostrados na tabela 2 foram obtidos utilizando a API de mapas do Google para direções, que permite entrar com uma origem e um destino e uma lista de paradas em formato de latitude e longitude. O tempo mostrado não leva em consideração o período em que o caminhão coletor ficará estacionado para realizar o recolhimento dos resíduos dos pontos coletáveis.

V. CONCLUSÃO

Algoritmos envolvendo Problema de Roteamento de Veículos são frequentemente usados em aplicações voltados para otimização de rotas e para o desenvolvimento de inovações voltados para sistemas IoT. O algoritmo implementado propõe uma solução para traçar rotas que recebe como entrada a origem, o destino, uma lista de caminhões contendo sua capacidade e uma lista de grafos por região contendo os pontos de coleta (nós do grafo) que combina as heurísticas: orientada a tempo e do vizinho mais próximo.

Para o cenário proposto foi feita a análise dos pontos aptos para coleta, a divisão desses pontos em regiões geográficas, geração do grafo, onde seus nós são os pontos coletáveis e as arestas são as distâncias entre cada nó e os demais pontos. O algoritmo dividiu os 30 pontos em 5 regiões geográficas e traçou 9 rotas com caminhões com capacidade de 500 litros para recolher 4382 litros de resíduos em um tempo que varia de 38 a 87 minutos.

Para trabalhos futuros, pretende-se incluir para a soma do tempo total em que o caminhão ficará parado em cada ponto, o que pode ser feito através de uma média probabilística que dependerá de aspectos físicos de como será feita a coleta nos pontos coletáveis, além da utilização de algoritmo genético para a solução do problema de roteamento.

VI. REFERÊNCIAS BIBLIOGRÁFICAS

CHEN, Shanzhi, et al. A Vision of IoT: Applications, Challenges, and Opportunities With China Perspective. IEEE, 2004.

ARAKAKI, Reinald. Monografia O PROBLEMA DE ROTEAMENTO DE VEÍCULOS E ALGUMAS METAHEURÍSTICAS. Exame de Qualificação do Curso de Computação Aplicada – CAP. 1998.

TILK, Christian et al. Nested branch-and-price-and-cut for vehicle routing problems with multiple resource interdependencies. European Journal of Operational Research, 2019.

TALARI, Saber, et al. A Review of Smart Cities Based on the Internet of Things Concept. Portugal, 2017.

SOLOMON, Marius. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. Massachusetts, 1987.

BAKER, Barrie. A Genetic Algorithm for the Vehicle Routing Problem. Reino Unido, 2003.

MOHAMMED, Mazin. Solving Vehicle Routing Problem by Using Improved K-Nearest Neighbor Algorithm for Best Solution. Journal of Computational Science, 21, 232-240. 2017.

SILVA, Gabriel et al. Algoritmos Heurísticos Construtivos Aplicados ao Problema do Caixeiro Viajante Para Definição de Rotas Otimizadas, 2013.

VII. AGRADECIMENTOS

Este trabalho conta com o apoio da Samsung Eletrônica da Amazônia Ltda. através da Lei de Informática, AGIN - Agência de Inovação da UEA e FUEA - Fundação Universitatis de Estudos Amazônicos.

VIII. COPYRIGHT

Direitos autorais: Os autores são os únicos responsáveis pelo material incluído no artigo.