Universidade Presbiteriana Mackenzie

Meta-Heurísticas e Técnicas de Otimização Aplicadas

Projeto Final

Aplicação do Algoritmo Genético na Resolução do Problema do Caixeiro Viajante como Continuação do Estudo do Problema da Localização de Instalações

Samuel Willian Alves Wu

Prof. Dr. Arnaldo R. A. Vallim Filho

06/2023 - São Paulo

**1. Introdução**

Este artigo utiliza como base o trabalho anterior que teve como objetivo estudar métodos de otimização combinatória para resolver o problema de localização de instalações (Samuel, Vallim, 2021). Inicialmente, tinha-se a intenção de estudar métodos alternativos para a etapa de clusterização e definição de centróides. Entretanto, optamos por abordar uma etapa posterior ao problema da localização de instalações, o de roteirização de veículos. Como um passo inicial para a resolução desse problema, que abordaria questões mais profundas como a topografia de determinado tipo de terreno, custo de transporte, tempo de trânsito, etc, abordamos o problema do caixeiro viajante.

O problema do caixeiro viajante (Traveling Salesman Problem ou TSP) descreve a situação de um viajante que precisa passar por N cidades, sendo necessário passar uma vez por cada cidade, retornando ao ponto inicial no final do processo. O objetivo do problema é atender essa condição otimizando o percurso de tal forma a minimizar a distância percorrida. Com o aumento do número de variáveis, ou seja, as cidades a percorrer, o problema cresce de forma exponencial, se configurando como NP-difícil e não há algoritmos capazes de encontrar a solução ótima em um tempo polinomial (Yong Shi e Yuanying Zhang, 2022).

Sendo assim, a função objetivo, ilustrada na imagem 1, deve minimizar a somatória de todos os trajetos entre as cidades, sujeito a ter apenas um arco como ponto de início e um como ponto final.

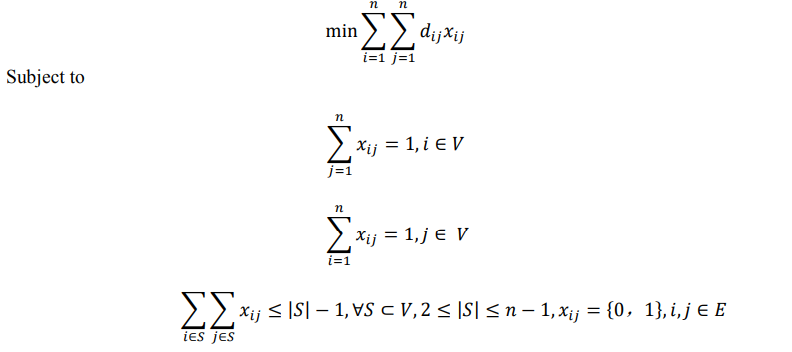


Imagem 1 - Modelagem matemática para o TSP

2. Revisão Bibliográfica

Abdulaziz Alorf (2023) em um estudo recente comparou diversas metaheurísticas com o objetivo de entender a evolução dos estudos nessa área, podendo também recomendar melhores opções para a comunidade. Problemas de otimização NP-difícil são considerados intratáveis por abranger espaços de busca grandes e complexos. Algoritmos exatos não são capazes de encontrar as soluções ótimas, escalando o tempo de execução de forma exponencial sendo que não há a certeza de encontrar a melhor solução. Sendo assim, algoritmos de aproximação como o algoritmo guloso oferecem uma abordagem mais eficaz, mesmo oferecendo soluções quase ótimas.

O termo metaheurística teve sua primeira aparição na pesquisa de Glover(1986) e comprime as palavras *meta* e *heurística*. O termo heurística deriva do termo grego heuriskein que tem como definição “encontrar”. Neto et al. (2010) dá ao termo heurística o significado de um conhecimento circunstancial, ou seja, compreendendo a etapa de uma camada superior do processo de descoberta de uma solução para um determinado problema. Um ramo de heurísticas tem como foco as buscas locais, partindo de uma solução gerada randomicamente e depois aprimorando essa solução de forma iterativa (Bianchi et al., 2008). Metaheurísticas são algoritmos que combinam heurísticas, que geralmente focam em um problema específico, em um framework geral que as fazem mais independentes de seus problemas.

Existem metaheurísticas de diferentes formas de implementação e foco, podendo ser categorizadas em muitas dimensões. Alguns exemplos são as metaheurísticas inspiradas na natureza, como a de otimização de colônia de formigas (ACO); as inspiradas em processos físicos, como a simulated annealing (SA); e as baseadas em população, como a de algoritmo genético (GA). Abdulaziz Alorf aponta o GA como uma metaheurística fundamental e em estado da arte, proposta inicialmente por Fraser em 1957 e formalizada por Holland em 1975.

O GA opera utilizando a metáfora da genética em populações (Corloni, A et al., 1996). Comumente um problema de otimização é traduzido como o problema de encontrar o indivíduo mais apto entre uma população de soluções. A aptidão é mensurada por uma função, que está ligada à função objetivo do problema a ser solucionado. Indivíduos são equivalentes a soluções e populações são um agrupamento de N indivíduos. Após a seleção de indivíduos, modificações ocorrem seguindo um conjunto de regras genéticas para haver a recombinação no qual um operador de cruzamento seleciona randomicamente 2 indivíduos, os recombinando para gerar 2 filhos. O operador de mutação age após a etapa de recombinação, tendo como parâmetro o resultado obtido na etapa anterior gerando uma nova população no qual alguns dos indivíduos sofreram mutações. A etapa de mutação introduz variações que garantem a possibilidade de explorar todo o espaço de busca, independente da população inicial estabelecida.

Como apontado por Yong Shi e Yuanying Zhang (2022), o problema do caixeiro viajante (Traveling Salesman Problem ou TSP) é um conhecido problema de otimização combinatória classificado como NP-Difícil, no qual um viajante tem o objetivo de visitar um conjunto de N cidades e ao final do processo, voltar para o ponto inicial. É necessário minimizar a distância total percorrida pelo viajante. É possível solucionar o problema por meio de algoritmos exatos e algoritmos aproximados. Entretanto, com o aumento do número de pontos a serem visitados, a solução ótima e a complexidade do algoritmo crescem explosivamente.

Para resolver o TSP de forma otimizada em um espaço de busca grande, o GA surge como uma opção que pode resolver este e qualquer tipo de problema prático (Jia Xu, Lang Pei, Rong-zhao Zhu, 2018). Entretanto, a implementação tradicional do GA possui alguns defeitos no processo de recombinação e na probabilidade fixa de mutação, sendo que se a probabilidade de mutação for baixa demais, o algoritmo irá convergir muito rápido e facilmente cairá em um local ótimo de soluções, diminuindo a diversidade da população.

Estudos surgiram para resolver o problema da falta de diversidade na população gerada pelo GA. Xiong et al. () apresentou métodos adaptativos para taxa de variação individual e o número de população. A estratégia consiste em manter uma baixa taxa de evolução para indivíduos mais aptos, enquanto indivíduos menos aptos são submetidos a uma taxa de mutação maior.

3. Desenvolvimento do Estudo

3.1 Coleta de dados

Este trabalho tem como base um estudo realizado anteriormente (Samuel, Vallim, 2021) que teve como objetivo utilizar técnicas de otimização combinatória para o problema de localização de instalações. Os dados utilizados do estudo anterior são reaproveitados neste artigo e são pertencentes a uma base pública proposta por Uchoa et al. (2017) e contém 100 conjuntos de dados, variando entre 101 e 1001 pontos. Os exemplares são compostos pelas variáveis cx e cy, representando sua localização em um plano e a variável quantity que representa a demanda dos pontos. Vale ressaltar que as bases são representativas e não correspondem a nenhuma operação logística real em particular.

Foram utilizados conjuntos de dados das grandezas de 200 pontos, 500 pontos e 1000 pontos, entretanto é necessário ressaltar que a implementação do algoritmo não roteiriza todos os pontos do conjunto, visto que há uma etapa anterior de análise de clusters.

Será usada a lista de pontos de demanda contendo sua localização, dada em cx e cy, e a qual facility o ponto está ligado. A lista de facilities também contém a localização dada em cx e cy.

3.2 Descrição do problema

O problema do caixeiro viajante é um problema combinatório clássico e é classificado como NP-Difícil, visto que com o aumento do número de pontos a serem visitados, o tempo computacional aumenta exponencialmente. Não existem algoritmos capazes de encontrar a solução ótima em um tempo polinomial (Yong Shi e Yuanying Zhang, 2022). É ilustrado que um viajante precisa visitar N cidades e voltar para o ponto inicial. Como realizar esse trajeto de forma a minimizar a distância total percorrida?

3.3 Modelagem de dados

Tendo a lista de pontos de demanda e a lista de facilities, é montado um vetor contendo as cidades, ligadas a uma facility especificada no algoritmo, e a facility que ficará na última posição do vetor. Não são usados todos os pontos do conjunto de dados na tarefa de roteirização, apenas os pontos ligados aos seus respectivos centroides. A seguir, utilizando o cálculo de distâncias euclidiano, montamos uma matriz quadrada contendo a distância entre cada elemento do vetor.

3.4 Modelagem matemática do problema

Yong Shi e Yuanying Zhang (2022) citam em seu estudo uma formula padrão para a modelagem matemática do problema (Imagem 1).

Sendo assim, o algoritmo desenvolvido implementa a função objetivo que recebe como parâmetro o percurso total e a matriz de distâncias, define o ponto inicial do percurso e calcula a somatória de todas as rotas entre os pontos (imagem 2).

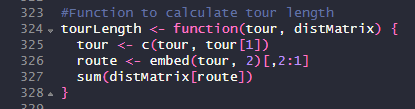


Imagem 2 - implementação da função objetivo

3.5 Implementação da metaheurística

Utilizamos a biblioteca “GA” do R para a implementação da metaheurística.

O processo se resume em implementar uma função de aptidão para definirmos como o processo de seleção irá acontecer, e chamar a função de montagem do algoritmo genético (imagem 3). A função tspFitness traduz o tour como uma solução e mensura sua aptidão pela seguinte lógica: quanto maior o tour percorrido, menor será o resultado retornado pela função, indicando soluções não ótimas.

A seguir, é definido o número de iterações, tamanho da população de soluções e a taxa de mutação.

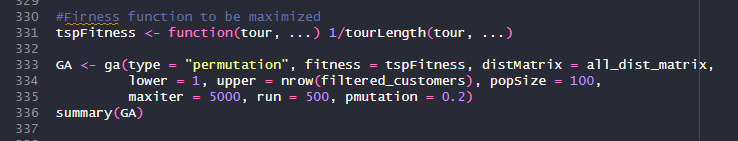


Imagem 3 - Implementação da função de aptidão e uso da metaheurística

3.6 Tratamento de resultados

Na última etapa, apresentamos o resultado de forma gráfica, indicando todos os pontos em um plano e traçando a rota percorrida por setas (imagem 4).

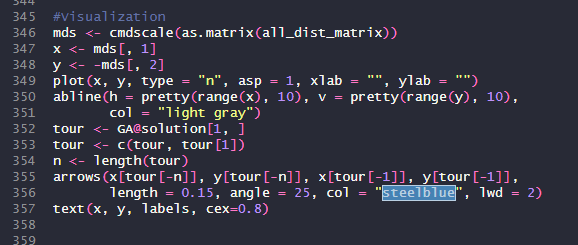


Imagem 4 - tratamento de resultados

4. Resultados e Discussão

Referências Bibliográficas

Yong Shi, Yuanying Zhang, The neural network methods for solving Traveling Salesman Problem, Procedia Computer Science, Volume 199, 2022