

Aplicação de UNSGA III para Projetos de Redes de Transporte Público Urbano

João Pedro Perez Resmer

Universidade Federal de Santa Catarina (UFSC)
Cx.P. 476 – 88040-370 Florianópolis – SC – Brasil
jpresmer@gmail.com

Rafael de Santiago

Departamento de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC)
Cx.P. 476 – 88040-370 Florianópolis – SC – Brasil
r.santiago@ufsc.br

RESUMO

Este estudo descreve a aplicação do algoritmo genético Unified Non-dominated Genetic Algorithm (UNSGA III) a um Urban Transport Network Design Problem (UTNDP). Nessa classe de problemas o objetivo é estabelecer rotas de transporte considerando pontos de acesso (estações onde as pessoas têm acesso ao transporte) e encontrar itinerários de veículos. O objetivo do estudo é avaliar o método em uma instância representada por grafo dirigido. A variação UNSGA foi escolhida em função da definição biobjetiva do problema. Essa abordagem utiliza seleção de progenitores por torneio, enquanto o NSGA convencional faz seleção aleatória. Para avaliar o desempenho do método proposto foram conduzidos experimentos em uma instância clássica para o problema. A partir disso foi possível observar os padrões de comportamento do método proposto com relação à seleção de indivíduos e as soluções geradas.

PALAVRAS CHAVE. Urban Transport Network Design Problem, Algoritmo genético, Seleção por Torneio.

MH – Metaheurísticas

ABSTRACT

This study describes the application of the genetic algorithm Unified Non-dominated Genetic Algorithm (UNSGA III) to an Urban Transport Network Design Problem (UTNDP). For this class of problems the goal is to create public transportation routes considering stops/access points (stations where people access public transportation), in addition they can solve fleet allocation and/or frequency problems. The goal is to evaluate the efficiency of the utilization of the algorithm on this type of problem in an instance represented by a directed graph. The UNSGA variation was chosen based on the bi-objective definition of the problem. This approach utilizes tournament selection, whilst the conventional NSGA goes with random selection. In order to evaluate the performance of the algorithm experiments were conducted with the algorithm on a well known instance. Based on the experiments it was possible to notice patterns related to the selection of individuals and the solutions found.

KEYWORDS. Urban Transport Network Design Problem, Algoritmo genético, Seleção por Torneio.

MH – Metaheuristics

1. Introdução

Endereçar problemas de transporte público pode reduzir o número de veículos, poluição e promover o bem-estar social nas cidades. De acordo com WHO [2019], o transporte público eficiente pode ajudar a reduzir as concentrações de poluição em áreas urbanas, baixar as taxas de acidentes, baixar os níveis de poluição sonora e melhorar a equidade no acesso de pessoas sem carro.

Especialmente problemas de projeto de redes de transporte urbanas (UTNDP, do inglês Urban Transport Network Design Problem) podem cobrir aspectos importantes do transporte público, como o projeto da rede atendida, frequência de atendimento nas rotas, agendamento dos horários de serviços e condutores. Geralmente consideram a malha viária, localização dos pontos de acesso e demanda para definir soluções ideais para atender a expectativa de usuários e operadores do serviço de transporte público John et al. [2014]. De acordo com Magnanti e Wong [1984], UTNDP é considerado NP-Difícil, o que o torna impraticável de resolver instâncias de tamanho real. Portanto o estudo de métodos heurísticos é importante para permitir que ferramentas possam estar disponíveis para tomadores de decisão.

Esse artigo de relata os resultados de uma pesquisa sobre a resolução de uma variante do problema UTNDP, que visa a otimização de rotas para atendimento das demandas em transporte público de qualquer tipo de modal. Como contribuições do artigo, destacam-se:

- A proposta de um modelo computacional evolutivo UNSGA III para atacar problemas da classe UTNDP em grafos dirigidos;
- A análise de resultados experimentais do método UNSGA III aplicado a um problema definido por um grafo dirigido.

O artigo segue a forma: a Seção 2 apresenta a definição do problema. A Seção 3 determina o método proposto. A Seção 4 trata da proposta dos experimentos e seus resultados. E, finalmente, a Seção 5 expõe considerações finais e algumas propostas para trabalhos futuros.

2. Definição do Problema

A variante do problema UTNDP endereçada no artigo pode ser definida a partir da seguinte forma: considere um grafo dirigido e fortemente conexo $G = (V, E)$, no qual V é o conjunto de ponto de acesso às rotas de ônibus, E é o conjunto de vias entre os pontos de acesso, uma matriz Θ o tempo requerido para viajar entre os pontos de acesso, e uma matriz de demanda D , que mapeia a demanda dos usuários (origem-destino).

2.1. Definição de Linha de Ônibus

Dada a natureza do problema, pelas propriedades dos grafos dirigidos, não há garantia de que existe um caminho simétrico à rota de uma linha de ônibus por onde se possa retornar ao ponto de partida. Por isso, define-se uma linha de ônibus por:

- Uma frota de tamanho f ;
- Uma rota principal com vértice de partida v e uma lista L de vértices intermediários ordenada de forma que seja u_i o antecessor de u_{i+1} na lista L a aresta (u_i, u_{i+1}) é percorrida na rota da linha. Sendo esta rota circular ou não;
- Uma rota de retorno de definição análoga a rota principal, de forma a ligar o ponto de partida e o último vértice da rota principal, se necessário (Essa lista pode ser vazia).

2.2. Características de Abordagem do Problema

Assume-se que a instância do problema apresente ao menos as propriedades baseadas em Mumford [2013] definidas a seguir:

- O número de vértices presentes na rota principal de uma rota nunca extrapola um limite predefinido (MAX VERTICES);
- O número de vértices presentes na rota principal de uma rota nunca é inferior a um limite mínimo predefinido (MIN VERTICES);
- Os valores de demanda entre os vértices do grafo G permanecem os mesmos durante todo o período de observação;
- Existe um valor de penalidade (TRANSFER PENALTY) para a troca de rotas dentro de um conjunto de rotas ao percorrer um caminho entre vértices v e u quaisquer;
- Existe um valor de penalidade (UNREACHABLE STOP PENALTY) caso dado um conjunto de rotas R não seja possível alcançar um vértice u qualquer a partir de um v qualquer;
- O modelo não considera a frequência dos veículos diretamente. Apenas distribui uma frota total f entre as rotas baseando-se na demanda;
- As escolhas dos passageiros por rotas se baseiam no menor caminho (considerando penalidades de transferência).

3. Método Proposto

O método proposto é uma implementação de UNSGA III que está especificado nas subseções a seguir.

3.1. Preparação dos Dados de Entrada

Os dados são retirados de três arquivos que contém, respectivamente, coordenadas, custos das arestas e a matriz de demanda, a partir desses dados é gerado o grafo G e uma matriz de demanda D . Com o grafo G e a matriz D criados é possível gerar uma população inicial com base em uma série de parâmetros e dar prosseguimento a execução.

3.2. Geração da População Inicial

O algoritmo de geração gera p conjuntos de rotas de tamanho entre r e $1.15 \cdot r$ onde cada rota tem uma rota principal de tamanho contido no intervalo $[\text{MIN VERTICES}, \text{MAX VERTICES}]$. Primeiramente, durante a geração de uma rota é escolhido um vértice inicial de forma aleatória. Em seguida, os demais vértices são selecionados, também de forma aleatória, respeitando a vizinhança do seu antecessor. Após isso, é encontrada uma rota de retorno usando uma busca em largura (Breadth First Search, BFS). Após a geração de um conjunto de rotas este é avaliado e caso necessário passa por uma função de reparo. O conjunto só é adicionado a população inicial se todos os vértices em V estiver contido em ao menos uma rota.

3.3. Funções Objetivo

A função do custo do operador é dada pela equação (1), onde $\alpha(r)$ corresponde ao número de vértices atingidos por uma rota r e $\omega(r)$ ao tamanho da frota disponível para a rota r . O custo do passageiro é definido pela função (2), onde D_{ij} é o valor da demanda entre os vértices i e j e $\Theta_{ij}(R)$ o tempo mínimo entre i e j , já considerando penalidades, para o conjunto de rotas R .

$$Co(R) = \sum_{i=1}^n \frac{\alpha(r_i)}{\omega(r_i)} \quad (1)$$

$$Cp(R) = \frac{\sum_{i,j=1}^n D_{ij} \Theta_{ij}(R)}{\sum_{i,j=1}^n D_{ij}} \quad (2)$$

O custo do operador é calculado de forma simples já que basta acessar o tamanho da lista de vértices das rotas e número de veículos vinculados a cada uma delas. Já para o cálculo do custo do passageiro é utilizada uma variação do algoritmo de Floyd-Warshall. Essa variação identifica transferências entre rotas para encontrar os valores D_{ij} e só então é calculado o valor $Cp(R)$ com base na Enquação (2).

3.4. Utilização do UNSGA III

O algoritmo escolhido foi o UNSGA III definido de acordo com o Algoritmo 1. A escolha foi feita pois a partir dos experimentos realizados em Seada e Deb [2014] entende-se que essa versão do algoritmo tende a apresentar melhor desempenho que a versão NSGA II dado o problema.

O UNSGA III parte de uma população inicial e subsequentemente gera populações a partir da atual e seleciona os indivíduos que devem seguir para a próxima iteração. Para cada geração, a população selecionada anterior é unida aos descendentes gerados a partir dela. Após a união dos grupos os valores referentes as funções objetivo para cada uma das soluções é calculado. Em seguida os indivíduos são agrupados de acordo com as relações de dominância¹ Então é feita a primeira seleção dos indivíduos. Com a normalização dos valores das funções objetivo e a associação dos indivíduos já selecionados aos pontos de referência em seguida. Uma segunda seleção é feita caso o número de indivíduos selecionados não seja suficiente. E por fim é gerado o grupo de descendentes da atual população. A criação do conjunto de descendentes é feita ao fim da iteração pois o método de comparação dos progenitores usado na escolha por torneio utiliza informações quanto a dominância, ranques e pontos de referência como critério.

A normalização dos valores dos objetivos é feita com base no método de Maximum of Non-dominated Front (MNDF) descrito em Blank et al. [2019]. A seleção dos progenitores durante a geração de descendentes é feita conforme o Algoritmo 2, seguindo os seguintes critérios:

- Se ambos os candidatos estão associados ao mesmo ponto de referência, então o indivíduo de menor ranque é selecionado. Caso os ranques sejam os mesmo o indivíduo que estiver a menor distância do ponto de referência é selecionado.
- Caso os candidatos não estejam associados ao mesmo ponto de referência, então o progenitor é escolhido de forma aleatória.

A seleção dos indivíduos é feita com base nas frentes de dominância que são o resultado, junto aos ranques, do algoritmo de Non-dominated Sort. Este algoritmo separa os indivíduos em frentes com base nas relações de dominância, como definido no Algoritmo 3. Caso esse critério seja insuficiente é aplicado o algoritmo de Niching. Este algoritmo é baseado em eixos de referência que podem ser predefinidos ou variar por iteração. A alternativa escolhida foi usar eixos predefinidos, os eixos são determinados pela origem e os vetores [4 0], [4 1], [4 2], [4 3] e [4 4]. Baseando-se nesses eixos o algoritmo associa o indivíduo mais próximo de cada eixo e o inclui na população. Para isso, parte-se do eixo com menos indivíduos já selecionados associados até alcançar o tamanho parametrizado de população.

4. Experimentos

Foram propostas duas rodadas de experimentos. Para a primeira rodada o algoritmo foi executado na instância de teste Mumford0, utilizado em Mumford [2013]. Nessa rodada foram executadas repetidamente (9 execuções) algumas combinações com as seguintes possibilidades de parâmetros:

¹Um indivíduo domina um outro quando desempenha melhor em pelo menos um dos objetivos e é ao menos equivalente nos demais.

Algorithm 1: UNSGA III

```
1  $H \leftarrow \text{Hyperplane}(2)$ 
2  $\text{Offspring} \leftarrow \text{list}()$ 
3  $\text{Associations} \leftarrow \text{null}$ 
4  $l \leftarrow \text{length}(\text{Population})$ 
5  $\text{generation} \leftarrow 0$ 
6 while  $\text{generation} < \text{MaxGenerations}$  do
7    $r \leftarrow \text{Population} \cup \text{Offspring}$ 
8    $\text{ObjValues} \leftarrow \text{list}()$ 
9   for each  $i$  in  $r$  do
10     $\text{ObjValues}_i \leftarrow [\text{Co}(i), \text{Cp}(i)]$ 
11  end
12   $\text{Fronts}, \text{Ranks} \leftarrow \text{NonDominatedSort}(\text{ObjValues})$ 
13   $S \leftarrow \text{list}()$ 
14   $i \leftarrow 0$ 
15  while  $\text{length}(S) + \text{length}(\text{front}) \neq \text{length}$  do
16     $S \leftarrow S \cup \text{Fronts}_i$ 
17     $i \leftarrow i + 1$ 
18  end
19   $K \leftarrow l - \text{length}(S)$ 
20   $\text{UpdatePlane}(H)$ 
21   $\text{Normalize}(\text{ObjValues}, H)$ 
22   $\text{RemainingFronts} \leftarrow \text{Fronts} - S$ 
23   $\text{Associations} \leftarrow \text{Associate}(\text{RefPoints}, S)$ 
24  if  $K > 0$  then
25     $S \leftarrow \text{Niching}(S, \text{RefPoints})$ 
26  end
27   $\text{Offspring} \leftarrow \text{GenerateOffspring}(S, \text{Ranks}, \text{Fronts}, \text{RefPoints})$ 
28 end
```

Algorithm 2: Seleção de Pais

```
1 Pick  $\leftarrow$  null
2 Parent1ReferencePoint  $\leftarrow$  DetermineRefPoint(Parent1, Associations)
3 Parent2ReferencePoint  $\leftarrow$  DetermineRefPoint(Parent2, Associations)
4 if Parent1ReferencePoint = Parent2ReferencePoint then
5   if Parent1Rank > Parent2Rank then
6     Pick  $\leftarrow$  Parent2
7   else if Parent1Rank < Parent2Rank then
8     Pick  $\leftarrow$  Parent1
9   else
10    Distance1  $\leftarrow$  CalculateDistance(Parent1, Parent1ReferencePoint)
11    Distance2  $\leftarrow$  CalculateDistance(Parent2, Parent2ReferencePoint)
12    if Distance1 < Distance2 then
13      Pick  $\leftarrow$  Parent1
14    else if Distance1 < Distance2 then
15      Pick  $\leftarrow$  Parent2
16    else
17      Pick  $\leftarrow$  RandomChoice(Parent1, Parent2)
18    end
19  end
20 else
21   Pick  $\leftarrow$  RandomChoice(Parent1, Parent2)
22 end
23 return Pick
```

Algorithm 3: Non-dominated Sort

```
1 Para dois objetivos
2 Individual1Co  $\leftarrow$  Co(Individual1)
3 Individual1Cp  $\leftarrow$  Cp(Individual1)
4 Individual2Co  $\leftarrow$  Co(Individual2)
5 Individual2Cp  $\leftarrow$  Cp(Individual2)
6 return Individual1Co  $\leq$  Individual2Co and Individual1Cp  $\leq$  Individual2Cp
   and (Individual1Co < Individual2Co or Individual1Cp < Individual2Cp)
```

- Penalidade de Transferência de Rota: 15, 20, 25;
- MIN VERTICES: 4, 3;
- MAX VERTICES: 8, 10;
- Probabilidade de Mutação: 10%, 25%, 50%;
- Penalidade por Vértice Inalcançável: 120;
- Tamanho da Frota: 15;
- Número Mínimo de Rotas: 6;
- Tamanho da População: 200;
- Número de Gerações: 200.

Nos resultados da primeira rodada foi possível observar que os melhores valores foram encontrados, em média, surgiram das combinações (Probabilidade de Mutação, MAX VERTICES, MIN VERTICES, Penalidade de Transferência de Rota, Penalidade por Vértice Inalcançável) $a = (50\%, 8, 3, 15, 120)$ e $b = (50\%, 8, 4, 15, 120)$.

A partir disso percebe-se uma predominância dos valores: 15 para Penalidade de Transferência de Rota e 8 para MAX VERTICES entre os parâmetros que apresentaram as melhores médias. Além disso ambas as melhores parametrizações médias apresentam 50% de probabilidade de mutação, ainda que algumas parametrizações com 10% de probabilidade, fortemente reduzida, também tenham apresentado bom desempenho. A variação entre 3 e 4 para o valor de MIN VERTICES se mostrou menos significativa. O melhor valor encontrado foi $Co = 22,37$ e $Cp = 42,7$ sob os parâmetros $e = (50\%, 8, 4, 15, 120)$. O pior valor encontrado foi $Co = 34,65$ e $Cp = 49,43$ com parâmetros $f = (10\%, 10, 4, 20, 120)$.

Essa constatação corrobora com a realizada a partir do estado médio quanto ao efeito dos parâmetros no desempenho da execução. Já que uma das parametrizações de melhores valores médios também teve o melhor desempenho individual. O desempenho dos parâmetros f também indica um favorecimento do valor de 50% para probabilidade de mutação em relação ao valor de 10%. Os demais valores de média se podem ser observados na Tabela 1.

Foi possível observar também a tendência do comportamento do algoritmo quanto aos indivíduos preservados. O algoritmo ainda tende a preservar indivíduos em um formato hiperbólico aproximando-se dos eixos x e y como pode-se observar na Figura 1. Isso representa a preservação dos indivíduos de melhor desempenho. Ainda assim, o algoritmo preserva também indivíduos que se aproximam dos eixos de referência o que aumenta a variabilidade entre indivíduos.

Já para a segunda rodada foram executadas 100 repetições com os parâmetros inicial, sendo esses o primeiro valor de cada parâmetro citado. Também na instância Mumford0. A partir dessa segunda rodada de execuções é possível observar o comportamento do algoritmo em relação a sua complexidade de tempo. A complexidade de tempo do algoritmo é influenciada principalmente pela função que calcula o custo do passageiro que é uma variação de Floyd-Warshall e portanto tem complexidade $O(n^3)$, onde n é o número de vértices do grafo de entrada. Porém há um fator de variabilidade considerável devido ao tamanho variável das rotas e a presença de escolhas aleatórias. Pois essas escolhas podem ser decididas como impróprias e demandar maior número de iterações.

Isso ocorre no algoritmo de mutação. A ocorrência de mutação é dependente de uma probabilidade passada por parâmetro. Uma vez que se decida pela mutação o número de alterações

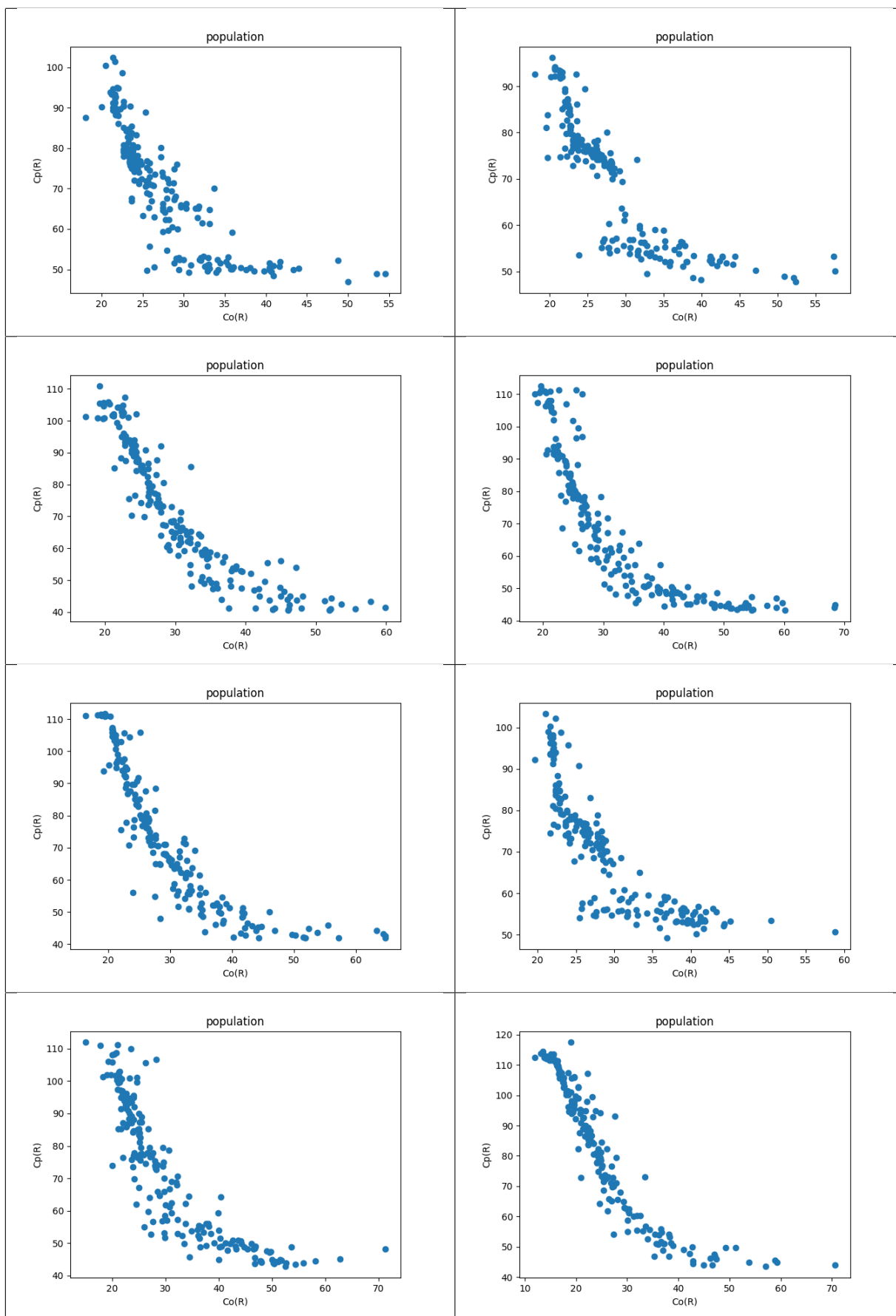


Figura 1: Valores dos Objetivos na População ao Fim da Execução

Tabela 1: Parâmetros de Teste e Valores por Objetivo da Melhor Solução

Prob. de Mutação	MAX VERTICES	MIN VERTICES	Penalidade de Transf.	Co	Cp
10%	8	4	20	24,75	50,83
25%	8	4	20	24,97	52,26
25%	8	4	20	24,17	58,23
50%	8	4	20	24,62	52,01
25%	10	4	20	25,55	60,24
50%	10	4	20	26,84	53,26
25%	10	3	20	24,52	56,06
50%	10	3	20	23,96	56,72
25%	8	4	15	25,10	46,03
10%	10	4	15	(25,83	48,10
25%	10	4	15	24,37	52,13
50%	10	4	15	(24,52	49,08
25%	8	3	15	24,01	49,19
10%	10	3	15	22,59	52,81
25%	10	3	15	24,81	48,03
50%	10	3	15	22,82	52,90

é variável, sendo no pior caso n e no melhor caso 1. Porém ao início de cada alteração um vértice é escolhido aleatoriamente e caso a escolha seja imprópria a decisão deve ser tomada novamente. Além disso ao final de cada modificação pode ser necessário ou não recalcular o trajeto de volta por meio de BFS (complexidade $O(n + m)$), dependendo do tipo de modificação efetuada. Considere n e m são o número de vértices e arestas respectivamente.

Como é possível observar na Figura 2, a variação de tempo total foi expressiva. Com média de aproximadamente 884,44 segundos, variância de aproximadamente 716,63 e desvio padrão de aproximadamente 26,77. Esse desvio padrão representa aproximadamente 3,03% sob a média.

Essa rodada também serviu para observar os resultados obtidos com a repetição da execução de uma instância sob os mesmos parâmetros. Observou-se uma variabilidade considerável porém ainda considerável no custo do operador da melhor solução com média. Variância e desvio padrão de respectivamente 25,17, 3,91 e 1,98. Sendo esse valor de desvio padrão aproximadamente 7,86% da média. O custo do passageiro apresentou uma variabilidade levemente menor com valores de média, variância e desvio padrão de respectivamente 45,26, 10,65, 3,26. Sendo que esse valor de desvio padrão representa aproximadamente 7,21% da média. A melhor solução obtida apresentou valores $Co = 21,83$ e $Cp = 42,43$ para custo do operador e passageiro. E a população final da execução que gerou essa solução pode ser observada na figura.

5. Conclusões

Foi proposto um novo método para a otimização de transporte público multimodal utilizando um algoritmo genético sob uma instância representada de uma forma pouco explorada. Um grafo G dirigido e fortemente conexo foi utilizado. Foram conduzidos experimentos para análise de possíveis parâmetros e os resultados foram disponibilizados e discutidos no presente artigo.

A partir dos resultados foi possível observar que as melhores soluções apresentam valores mais equilibrados. Mesmo que o algoritmo preservar indivíduos com resultados particularmente bons para um dos objetivos mesmo que esses apresentem mal desempenho em outro. As soluções de melhor desempenho costumam apresentar entre uma e três rotas mais longas e as demais rotas mais curtas com intersecções que ligam elas aos demais pontos de acesso.

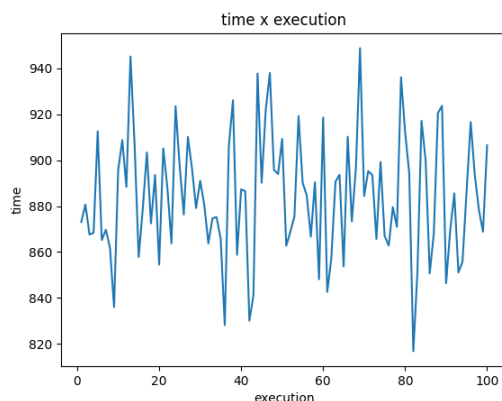


Figura 2: Tempo (s) X Execução

O método demonstrou bons resultados em relação a otimização alcançada na instância Mumford0. O algoritmo presa pela variabilidade nas populações e possibilita também o aumento do número de funções objetivo para definição da qualidade dos conjunto de rotas alcançados. Por isso, acredita-se no potencial da solução dado o refinamento necessário.

Como possíveis trabalhos futuros, destaca-se: (i) ampliar os experimentos para instâncias reais do problema; (ii) experimentar e analisar os resultados com um aumento no número de funções objetivo; e (iii) realizar experimentos com objetivo de identificar a relação entre o número de gerações e a variação de qualidade do melhor indivíduo encontrado.

Referências

- Blank, J., Deb, K., e Roy, P. C. (2019). Investigating the normalization procedure of nsga-iii. In *International Conference on Evolutionary Multi-Criterion Optimization*, p. 229–240. Springer.
- John, M. P., Mumford, C. L., e Lewis, R. (2014). An improved multi-objective algorithm for the urban transit routing problem. In *European Conference on Evolutionary Computation in Combinatorial Optimization*, p. 49–60. Springer.
- Magnanti, T. L. e Wong, R. T. (1984). Network Design and Transportation Planning: Models and Algorithms. *Transportation Science*, 18(1):1–55. ISSN 00411655.
- Mumford, C. L. (2013). New heuristic and evolutionary operators for the multi-objective urban transit routing problem. In *2013 IEEE congress on evolutionary computation*, p. 939–946. IEEE.
- Seada, H. e Deb, K. (2014). U-nsga-iii: A unified evolutionary algorithm for single, multiple, and many-objective optimization. *COIN report*, 2014022.
- WHO (2019). Health and sustainable development: public transport.