



REFINAMENTO DE SOLUÇÕES ÓTIMAS PARA O PROBLEMA DE ROTEAMENTO DE VEÍCULOS ESCOLARES

Júlio César de Oliveira Silva Júnior

Programa de Pós-Graduação em Engenharia Elétrica - PPGEE - UFMG
Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte - MG, 31270-901
julio.surflab@gmail.com

Rafael Frederico Alexandre

Universidade Federal de Ouro Preto - UFOP
R. 36, 115 - Loanda, João Monlevade - MG, 35931-008
rfalexandre@decea.ufop.br

Bruno Monteiro

Universidade Federal de Minas Gerais - UFMG
Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte - MG, 31270-901
bruno_ic_monteiro@hotmail.com

João Antônio de Vasconcelos

Universidade Federal de Minas Gerais - UFMG
Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte - MG, 31270-901
jvasconcelos@ufmg.br

RESUMO

Este trabalho tem por objetivo propor uma metodologia para o refinamento de rotas construídas por algoritmos de otimização como soluções para o problema de roteamento de veículos escolares (SBRP). A legislação de alguns estados brasileiros permite que alunos matriculados na rede pública de ensino possam caminhar uma distância limitada entre suas residências e seus pontos de embarque. Isto possibilita um reposicionamento de estudantes, o que permite que fragmentos de rotas sejam eliminados de soluções ótimas, refinando-as.

PALAVRAS CHAVE. Otimização. Roteamento. Refinamento.

Tópicos (L&T, TAG)

ABSTRACT

This work aims to propose a methodology to refine routes built by optimization algorithms as solutions for the School Bus Routing Problem (SBRP). The legislation from some Brazilian states allow students enrolled in the federal net of education to walk a limited distance from their home towards boarding points. The possibility of student repositioning allows elimination of routes' fragments from optimum solutions, refining them.

KEYWORDS. Optimization. Routing. Refinement.

Paper topics (L&T, TAG)



1. Introdução

O Problema de Roteamento de Ônibus Escolar, ou SBRP (School Bus Routing Problem) é um problema cuja solução visa planejar eficientemente um cronograma para que uma frota de veículos escolares recolha estudantes em vários pontos de embarque e os entregam em suas respectivas escolas, ou pontos de desembarque. Neste tipo de problema, várias restrições precisam ser satisfeitas, como a capacidade máxima dos veículos, o tempo máximo de permanência do estudante no veículo e a janela de tempo de uma escola (horário de início ou fim das aulas). De acordo com Park e Kim [2010], a solução de um SBRP engloba, geralmente, 5 sub-problemas: preparação dos dados, seleção de pontos de embarque, planejamento e desenho das rotas (viagens), sincronização com os horários das escolas e refinamento das rotas. A preparação dos dados tem como objetivo preparar as informações necessárias aos demais sub-problemas. O grafo, ou malha de possíveis rotas, é montado e tem associado quatro conjuntos de informações: estudantes (quantidade, localização, escola designada, turno), pontos de embarque (localização, estudantes designados), escolas ou pontos de desembarque (localização, estudantes matriculados, horários das aulas), veículos (capacidade, velocidade média) e as arestas que interligam a malha (distâncias de percurso ponto a ponto). O grafo, ou malha, é também conhecido como matriz Origem-Destino e armazena as menores distâncias entre os pares de nós (pontos de embarque e desembarque). A matriz OD pode ser construída usando um sistema de informação geográfica (GIS), como o QuantumGIS e o AutoCad Maps. Em Kim e Jeong [2009] é apresentada uma comparação de desempenho de vários algoritmos para resolução do problema do caminho mínimo e é desenvolvida uma abordagem para a geração aproximada da matriz OD.

O sub-problema de seleção de pontos de embarque procura selecionar um conjunto de pontos do grafo e lhes atribuir estudantes. Normalmente, estudantes que residem em zonas rurais tem sua própria residência assumida como um ponto de embarque. Entretanto, em zonas urbanas normalmente se planeja localizações de pontos até onde estudantes devem se deslocar a pé para esperar pelo transporte. Na maioria dos casos os pontos de embarque são fixos e o sub-problema se resume a alocar, de forma eficiente, estudantes a pontos de embarque. Em Park e Kim [2010] é apresentada uma lista de trabalhos que consideram este sub-problema.

O sub-problema de planejamento e desenho das rotas de veículos exige a construção das rotas até os pontos de embarque. Existem vários algoritmos especializados a resolver este sub-problema, entre os quais destacam-se pesquisas como Chapleau et al. [1985], Dulac et al. [1980], Newton e Thomas [1969], Bowerman et al. [1995] e outras recentes como Defryn e Sörensen [2017], Zhang et al. [2017] e Mohammed et al. [2017].

No sub-problema de refinamento de rotas, finalmente, especifica-se a hora exata do início e término de cada rota e se define uma cadeia de rotas executadas pelo mesmo veículo. Newton e Thomas [1974] desenvolveram um modelo multi-escola para determinar todas as rotas para um distrito escolar na qual assumiram períodos de tempo distintos entre aulas de diferentes escolas, Li e Fu [2002] aplicaram o Algoritmo de Lawler [1972] para gerar uma rota inicial e um esquema de refinamento em que os pontos de embarque das rotas maiores são movidos para as rotas menores, e Graham e Nuttle [1986] mostra comparações para diversas heurísticas produzidas para o refinamento de rotas escolares. Entretanto, não é comum na literatura métodos de refinamento que explorem o reposicionamento de pontos de embarque ao permitir que estudantes caminhem de um ponto a outro. Como mencionado anteriormente, este artifício normalmente é aplicado no sub-problema de seleção de pontos de embarque e, mais especificamente, em regiões urbanas ou mais densamente povoadas. Entretanto, em regiões rurais e outras menos populosas, soluções finais podem ser, ainda, refinadas ao se explorar características de rotas geradas, as quais só podem ser identificadas após a completa geração de uma rota.

Este trabalho tem por objetivo propor uma nova metodologia de refinamento de rotas que explore a possibilidade de estudantes caminharem de um ponto de embarque a outro, principalmente em regiões rurais e pouco populosas. Este trabalho é dividido em 5 seções: a definição do



problema, onde se explica com maior detalhe o domínio onde a metodologia apresentada se insere, bem como sua ideologia básica; a descrição da metodologia, onde o algoritmo proposto é explicado em detalhes; um estudo de caso, onde é apresentado um exemplo de aplicação do algoritmo proposto e os resultados obtidos; a conclusão, onde o trabalho e seus resultados são brevemente resumidos e são feitas sugestões para futuros trabalhos nesta linha.

2. Definição do Problema

Quando um problema de roteamento de veículos escolares é abordado, normalmente é acompanhado de bases de dados que contém localizações geográficas ou euclidianas de nós (pontos de embarque e desembarque), e informações de arestas que os conectam para gerar um grafo. Entretanto, informações sobre as malhas viárias reais dificilmente são acessíveis, o que pode vir a criar deformações indesejáveis entre uma rota gerada por um algoritmo de otimização (rota baseada em um grafo) e sua aplicação prática (rota baseada em uma malha viária real). Considere a Figura 1(a) que representa uma rota ótima local obtida por um algoritmo de otimização cujo objetivo foi a minimização da distância total entre os pontos 1 e 9. Considere, ainda, que as únicas informações disponíveis sobre o grafo são: a localização dos pontos de embarque e desembarque (pontos escuros numerados de 1 a 9), o comprimento das arestas que os conectam e o sentido de tráfego de cada aresta. A Figura 1(a) apresenta uma rota coerente onde cada ponto é visitado apenas uma vez. Entretanto, quando a mesma rota é desenhada sobre uma malha viária real, como ilustrado pela Figura 1(b), a deformação causada pela topologia, configuração e forma das vias incorre na duplicação de certos trajetos devido à presença de *ramos viários* na malha. O trajeto $[2,3,4,5]$ da imagem 1, por exemplo, se torna $[2,\alpha,3,\alpha,4,\beta,5]$ quando avaliado sobre a 1(b). A sequência $[\alpha,3,\alpha]$ indica que um veículo precisou realizar um movimento de *ida* e *volta* sobre um mesmo caminho da malha, o qual será doravante nomeado *ramo viário*. A Figura 2 ilustra os ramos viários presentes na malha real da 1(b) e uma definição formal de *ramo viário* será apresentada na seção 3.

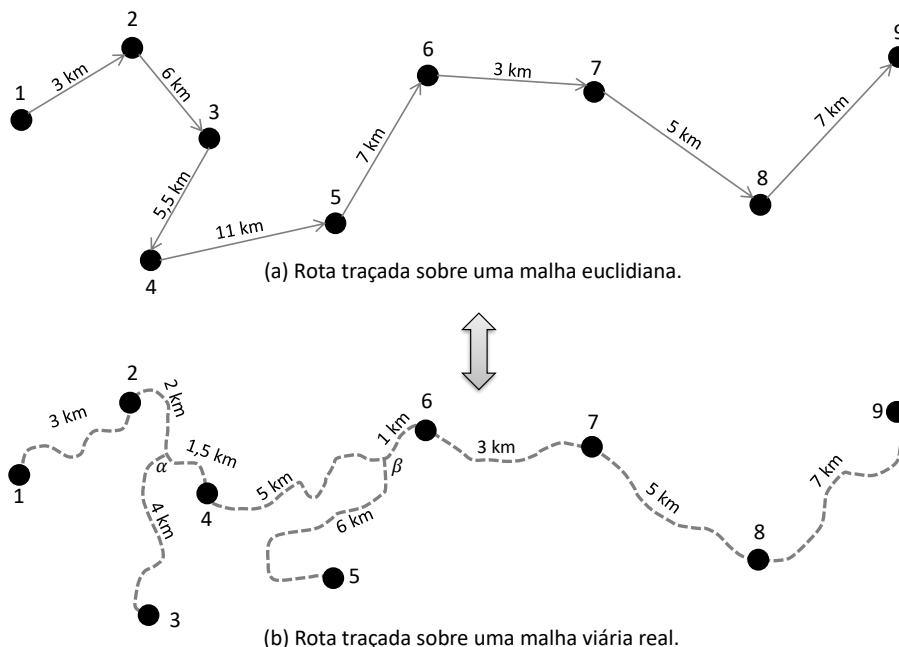


Figura 1: Exemplo da deformação entre uma rota ótima traçada sobre uma malha euclidiana, baseada em grafo, e a mesma rota traçada sobre uma malha viária real. As bifurcações da via são representadas por α e β .

Apesar de potencialmente inevitáveis do ponto de vista de otimização, a presença de *ramos viários* prejudica a otimalidade de uma solução. Porém, existem maneiras de se refinar trajetos já construídos de modo a reduzir ou até mesmo eliminar a presença destes ramos em problemas de

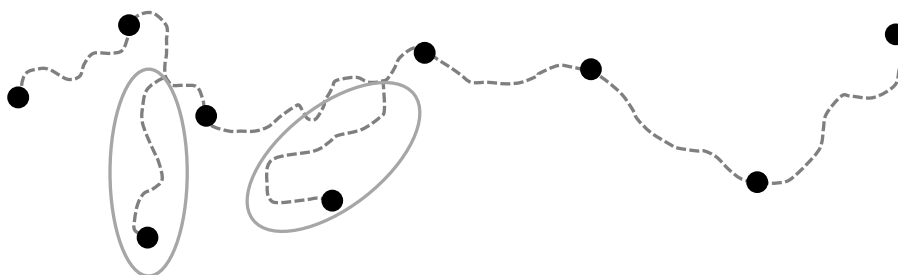


Figura 2: Exemplo de dois ramos viários sobre uma malha viária real.

roteamento de veículos escolares. Segundo a Portaria R36 da Secretaria de Estado da Educação do estado do Espírito Santo, Brasil (SEDU - ES), existe um limite para a distância que um estudante pode caminhar de sua residência até um ponto de embarque, e de um ponto de desembarque até sua escola. Assumindo que equivalente legislação seja recorrente em outros estados brasileiros, é possível explorar tal parâmetro de modo a, de posse de soluções ótimas pré-calculadas, reduzir ou eliminar ramos viários ao reposicionar estudantes de um ponto de embarque a outro.

A seguir será apresentada uma metodologia que permite refinar rotas ótimas geradas como soluções para o SBRP por meio da redução (ou eliminação) de ramos viários, redução esta que se dá pelo reposicionamento de estudantes de um ponto de embarque a outro, os quais podem se deslocar a pé por uma distância parametrizada.

3. Algoritmo proposto

O algoritmo proposto será apresentado em duas subseções dispostas a seguir. Primeiramente, alguns conceitos básicos precisam ser apresentados, seguidos da metodologia, a qual será explicada em detalhes.

3.1. Conceitos básicos

O algoritmo proposto é aplicável diretamente às soluções de um SBRP e independe dos métodos de otimização utilizados para gerá-las. Entretanto, é necessário que uma solução respeite algumas características, o que faz necessária a introdução de termos chave à aplicação desta metodologia. São eles:

- **Viagem:** O termo *Viagem* doravante se refere à rota ou percurso completo realizado por um único veículo em uma determinada solução. Uma solução deve ser composta por um conjunto de viagens e uma viagem deve ser composta por um conjunto de pontos de embarque P_i e de pontos de desembarque D_j . Tanto P_i quanto D_j podem se referir a pontos de ônibus ou a escolas. Pois, uma viagem pode ser resumida pela seguinte sequência:

$$Viagem = [P_1, P_2, P_3, D_1, P_4, D_2, D_3, P_2, P_4, \dots] \quad (1)$$

Os pontos de embarque e desembarque de uma viagem podem ser conectados por arestas (no caso de um grafo) ou, de forma equivalente, por vias reais de uma malha viária (respeitando-se topografia e contornos das vias).

- **Ramo viário:** Um ramo viário é constituído por um fragmento de rota sobre o qual um veículo transita duas vezes consecutivas em um movimento de *ida* e *volta*. Além disso, o ponto, ou nó, destino de um ramo viário deve ser de grau 1 (não pode possuir mais de uma aresta conectada a ele) e não pode ser um ponto de desembarque (uma vez que escolas não podem ser movidas de suas localizações originais). Um ramo viário não equivale, necessariamente, a uma aresta, como ilustra a Figura 3. Uma aresta é um percurso confinado entre dois nós, restrição que não se aplica a um ramo viário.

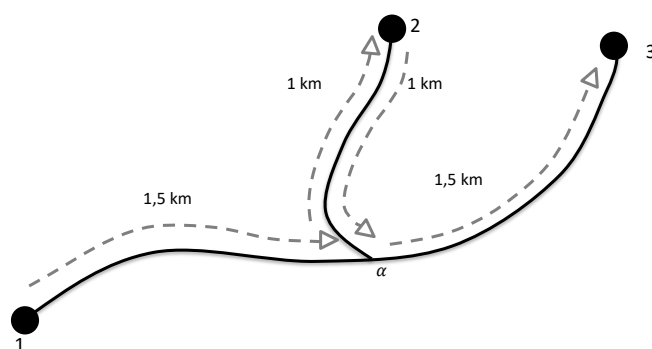


Figura 3: Exemplo de ramo em uma viagem definida pela sequência $[1, \alpha, 2, \alpha, 3]$. Assumindo que o ponto 2 é um ponto de embarque, um nó de grau 1 e α um vértice intermediário entre os pontos 1 e 2, o fragmento de 1 km ($\alpha \leftrightarrow 2$) que o tem como destino é definido como um ramo, uma vez que é necessário percorrê-lo duas vezes consecutivas em movimento de *ida* e *volta*. Note que o ramo em questão não se configura como uma aresta entre dois nós.

O algoritmo proposto visa reposicionar o máximo possível de pontos de embarque em todas as viagens que compõe uma solução, ao permitir que estudantes se desloquem a pé de uma posição inicial a um posição final vantajosa. Entretanto, uma vez que as viagens foram construídas por um método de otimização, não é uma tarefa fácil reposicionar pontos de rotas sem criar a necessidade de otimizá-las novamente. Justifica-se, pois, a eliminação de ramos viários, os quais levam, sempre, a pontos cujo reposicionamento não ameaça a otimalidade de uma viagem (a remoção de ramos tende, sempre a reduzir a distância total percorrida sem alterar a constituição da viagem - a sequência de pontos visitados e os alunos atendidos - de modo que a sequência ótima definida pelo indivíduo de otimização não é alterada; apenas seu comprimento é reduzido). Tomando-se como exemplo a Figura 3, o ramo viário de 2 km que leva ao ponto 2 pode ser eliminado caso se permita que alunos se desloquem a pé por uma distância de até 1 km. Como ilustrado pela Figura 4, a ausência do ramo viário desloca a posição do ponto 2 até o vértice α , e reduz a distância total percorrida pelo veículo em 2 km, o que representa 40% de economia para a viagem. Além disso, é possível verificar que o reposicionamento do ponto 2 não cria a necessidade de uma nova otimização.

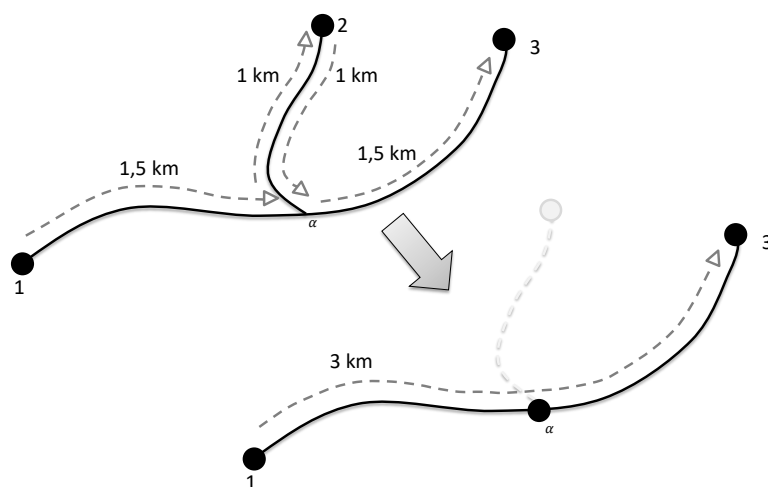


Figura 4: Resultado da eliminação de um ramo viário sobre um viagem definida pela sequência $[1, 2, 3]$. São reduzidos 2 km sobre 5 km de rota, o que configura 40% de economia para a viagem.

Apresentados os conceitos básicos, a metodologia será descrita em detalhes a seguir.



3.2. Metodologia

O algoritmo proposto foi dividido em três pseudocódigos para facilitar a compreensão. O Algoritmo 1 resume toda a metodologia, o Algoritmo 2 seleciona os ramos a serem removidos e chama o Algoritmo 3, o qual realiza a remoção. Cada um será apresentado e ilustrado separadamente.

Algoritmo 1: RESUMO DA METODOLOGIA

```
Input: Solução           /*Solução composta por um conjunto de viagens.*/
Output: NovaSolução
1 Begin
2 repeat
3   SoluçãoAnterior ← Solução
4   foreach Viagem ∈ SoluçãoAnterior do
5     NovaViagem ← Algoritmo 2(Viagem)
6     NovaSolução ← NovaSolução ∪ NovaViagem
7   end
8   Solução ← NovaSolução
9 until Avaliação da SoluçãoAnterior = Avaliação da NovaSolução
10 return NovaSolução
11 End
```

O Algoritmo 1 é responsável por receber uma solução, refiná-la, e devolver uma nova solução cuja avaliação é melhor que a da primeira. Portanto, para cada viagem que compõe uma solução, o Algoritmo 2 é chamado, o qual retorna uma nova viagem com todos os possíveis ramos eliminados. Esta nova viagem passa a compor a lista de viagens da nova solução. O processo é iterado até que as avaliações de ambas, solução e nova solução, sejam idênticas, o que significa que nenhum outro ramo viário pode ser eliminado. Neste caso, a rotina se encerra e a nova solução, refinada, é retornada.

O Algoritmo 2, o mais importante da metodologia, é responsável por selecionar os ramos viários de uma rota para serem eliminados. Ramos viários podem ser identificados por padrões na sequência de nós. Uma vez que estes são definidos como fragmentos de rota por onde um veículo transita duas vezes em sentido de *ida* e *volta*, um ramo viário confinado entre dois nós, A e B, apresentaria uma sequência de nós dada por A B A (visita-se o ponto A, segue-se para o ponto B e, como o ponto B é um nó de grau 1, retorna-se para o ponto A). Esta sequência configura um *padrão de remoção*. Entretanto, a definição empregada de ramo viário atesta que o mesmo não é constituído por uma aresta entre dois nós. A Figura 3 ilustra, inclusive, uma viagem a qual possui um ramo viário, mas cuja representação sequencial, [1, 2, 3], carece de qualquer padrão de remoção. Por este motivo é fundamental que sejam adicionados *pontos intermediários*, como por exemplo o ponto α indicado na Figura 3, à lista de pontos (ou nós) que compõe uma viagem. Doravante, um *ponto intermediário* será definido como:

- **Ponto intermediário:** Um ponto, ou nó, anteriormente inexistente tanto no grafo quanto na rota que compõe uma viagem, mas cuja localização geográfica ou euclidiana coincida com a localização de uma aresta (no grafo ou na malha euclidiana) ou via (na malha viária real).

Pontos intermediários são posições pontuais sobre arestas de um grafo (equivalentemente, sobre as vias de uma rota), e funcionam como pontos de segmentação. Enquanto pontos de embarque e desembarque são, costumeiramente, pontos de ônibus, residências ou escolas, um ponto intermediário é um acostamento de via. Pois, a ideia de segmentação de uma aresta/via deve vir acompanhada de uma interpretação plausível para os pontos intermediários construídos (uma posição que indicará a construção de um novo ponto de ônibus escolar, por exemplo), uma vez que haverá estudantes no local.



Algoritmo 2: SELEÇÃO DE RAMOS

```
Input: Viagem          /*A viagem é composta por um conjunto de pontos.*/  
Output: Viagem  
1 Begin  
2 foreach Ponto  $p \in$  Viagem do  
3   PontoAtual  $\leftarrow$  Ponto $p$   $\in$  Viagem  
4   PontoPosterior  $\leftarrow$  Ponto $p+1$   $\in$  Viagem  
5   if PontoAtual  $\neq$  PontoPosterior then  
6     ListaPontosIntermediários  $\leftarrow$  ListaPontosIntermediários  $\cup$  PontoAtual  
7     PontosIntermediários  $\leftarrow$  Todos os pontos intermediários entre PontoAtual e PontoPosterior  
8     ListaPontosIntermediários  $\leftarrow$  ListaPontosIntermediários  $\cup$  PontosIntermediários  
9     ListaPontosIntermediários  $\leftarrow$  ListaPontosIntermediários  $\cup$  PontoPosterior  
10  end  
11 end  
12 foreach PontoIntermediário  $p' \in$  ListaPontosIntermediários do  
13   PontoPivot  $\leftarrow$  PontoIntermediário  $p' \in$  ListaPontosIntermediários  
14   PontoPrimeiroPosterior  $\leftarrow$  PontoIntermediário  $p' + 1 \in$  ListaPontosIntermediários  
15   PontoSegundoPosterior  $\leftarrow$  PontoIntermediário  $p' + 2 \in$  ListaPontosIntermediários  
16   if PontoPivot = PontoSegundoPosterior and PontoPrimeiroPosterior não for uma escola then  
17     Algoritmo 3(PontoPrimeiroPosterior, PontoPivot, ListaPontosIntermediários, Viagem)  
18     if Viagem e ListaPontosIntermediários foram modificados pelo Algoritmo 3 then  
19       Reinicia loop desde o início  
20     end  
21   end  
22 end  
23 return Viagem  
24 End
```

A adição de pontos intermediários a uma viagem não é uma tarefa simples. Cada ponto intermediário precisa, necessariamente, ser um ponto sobre a rota gerada pelo algoritmo de otimização. Para SBRPs de malha euclidiana, esta tarefa pode ser mais simples. Entretanto, quando o problema é posto sobre uma malha viária real, pode ser bastante complexo. Mais à frente, neste mesmo trabalho, será apresentado um estudo de caso sobre malha viária real onde foi empregada a metodologia proposta, e a forma utilizada para se adicionar pontos intermediários será apresentada. Mas, como a forma de adição destes pontos depende da natureza da malha e da tecnologia utilizada para construí-la, não há como descrever ou sugerir um método único de adição. No momento, basta dizer que existem várias formas de se adicionar tais pontos e que eles são fundamentais para prosseguir com a metodologia proposta neste trabalho. A Figura 5 ilustra a adição de pontos intermediários sobre a viagem ilustrada pela Figura 3. Na Figura 5, foram adicionados 13 pontos intermediários, nomeados de acordo com o intervalo alfabético compreendido entre "a" e "m".

O Algoritmo 2 recebe uma viagem de entrada e retorna uma viagem modificada de saída. Para cada ponto que compõe a viagem de entrada, pontos intermediários entre o ponto em questão e seu sucessor, inclusive os próprios, são adicionados a uma lista de pontos. Então, para cada ponto intermediário na lista, três são elencados: o ponto em questão, seu sucessor e o seguinte. Estes três pontos são utilizados para procurar o padrão de pontos que define um ramo viário (ABA, por exemplo). Se um padrão é identificado, o Algoritmo 3 é chamado para realizar a remoção do ponto de grau 1. As Figuras 6 e 7 exemplificam a identificação de padrões de remoção.

O Algoritmo 3 é responsável pela remoção do ponto, ou nó, de grau 1 dentre os pontos que recebe como entrada. O ponto de grau 1 só pode ser removido caso ambas de duas condições sejam satisfeitas:

1. Se o ponto de grau 1 não for um ponto de desembarque.
2. Se a maior distância caminhada, dentre todas as distâncias caminhadas por todos os alunos

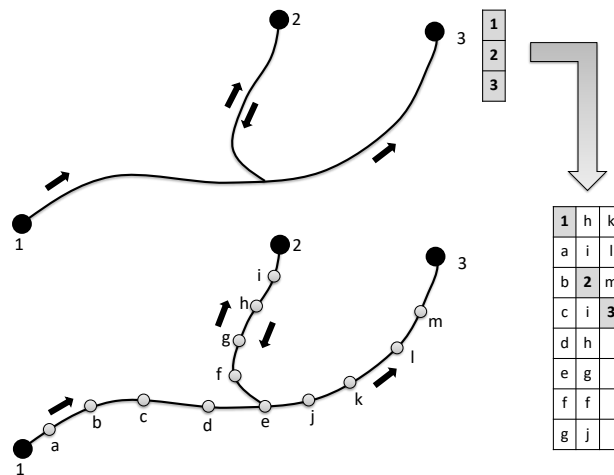


Figura 5: Adição de pontos intermediários em uma dada viagem de uma solução

Algoritmo 3: REMOÇÃO DE RAMOS

Input: PontoParaRemover, PontoParaManter, ListaPontosIntermediários, Viagem
Output: Input modificado por parâmetro

- 1 **Begin**
- 2 ComprimentoDoRamo \leftarrow Distância entre PontoParaRemover e PontoParaManter
- 3 DistanciaCaminhada \leftarrow Máxima distância já caminhada por algum estudante em PontoParaRemover
- 4 DistanciaMaxima \leftarrow Máxima distância que um estudante pode caminhar
- 5 **if** $DistanciaCaminhada + ComprimentoDoRamo \leq DistanciaMaxima$ **then**
- 6 PontoParaManter \leftarrow estudantes em PontoParaRemover
- 7 Viagem \leftarrow Viagem - PontoParaRemover
- 8 ListaPontosIntermediários \leftarrow ListaPontosIntermediários - PontoParaRemover
- 9 ListaPontosIntermediários \leftarrow ListaPontosIntermediários - PontoParaManter
- 10 **end**
- 11 **End**

presentes no ponto de grau 1, somada ao comprimento do ramo que será eliminado, seja igual ou inferior a um parâmetro *DistanciaMaxima*.

A condição 1 é verificada pelo Algoritmo 2. A condição 2, verificada pelo Algoritmo 3, garante que nenhum estudante, ao ser reposicionado, caminhe uma distância superior à permitida pela legislação do estado. Caso estas condições sejam satisfeitas, os alunos posicionados no ponto de grau 1 são reposicionados e o mesmo ponto é removido da viagem e da lista de pontos intermediários. O nó remanescente (aquele para o qual os estudantes foram reposicionados) também deve ser removido da lista de pontos intermediários, uma vez que este se repete no padrão de remoção e, caso não seja removido, incorrerá em duplicidade de dados. As figuras 6 e 7 exemplificam a remoção de ramos viários identificados.

4. Estudo de Caso

O estudo de caso realizado como forma de validação da eficiência do algoritmo proposto foi realizado sobre uma malha viária real, cujo grafo foi construído por meio do software comercial AutoCad Maps. Por meio desse software, é possível selecionar pontos manualmente sobre um mapa cartográfico real e localizar tanto pontos de embarque e desembarque quanto pontos intermediários (tantos quanto se queira). Ao se definir, via interface do AutoCad Maps, manualmente, a localização de um ponto, suas coordenadas são transferidas para o sistema gerenciador de banco de dados PostgreSQL; a partir deste banco, as coordenadas que definem um ponto (inclusive pontos intermediários) podem ser construídas, bem como as arestas que os conectam. Ou seja, as vias rurais reais são montadas manualmente via AutoCad Maps, o qual também é utilizado para construir

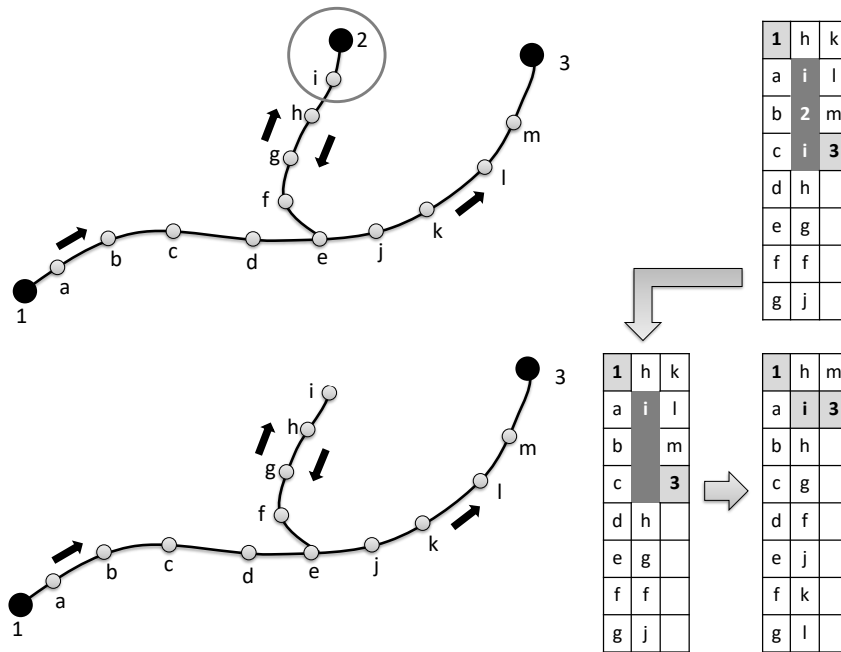


Figura 6: Identificação de um padrão de remoção e eliminação de um ramo da viagem

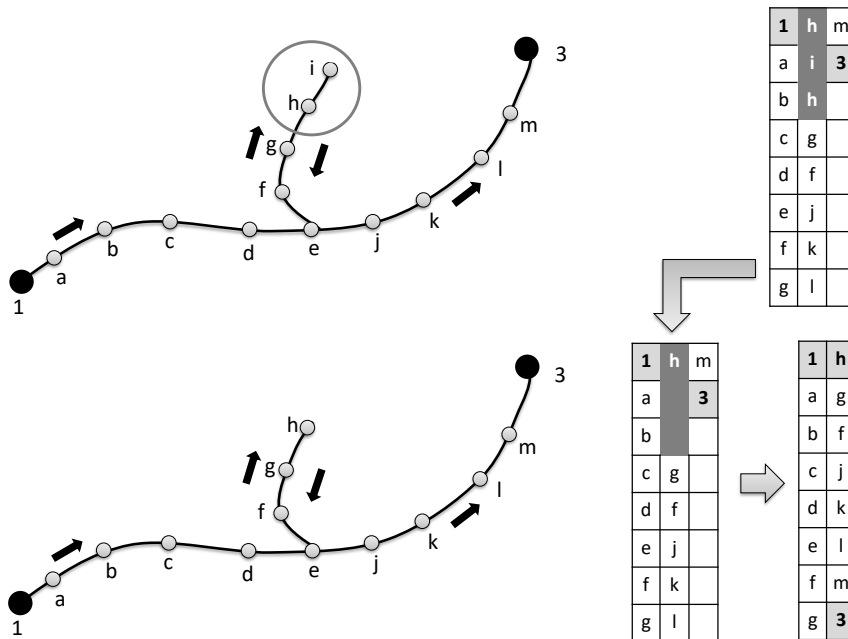


Figura 7: Identificação de um novo padrão de remoção e eliminação de um novo ramo da viagem

a informação de pontos de embarque, pontos de desembarque e pontos intermediários. Utilizamos um método de otimização de nossa autoria para construir uma rota ótima sobre uma malha construída para o município de Linhares, Espírito Santo, Brasil. Este método, o qual não será abordado neste artigo, utiliza uma meta-heurística híbrida baseada no Algoritmo de Seleção Clonal (CLONALG) e no método de agrupamento de dados, k -means++. A solução gerada foi impressa pelo Google Maps para gerar a imagem da Figura 8.



Figura 8: Exemplo de uma das viagens que compõem uma solução ótima para um SBRP.

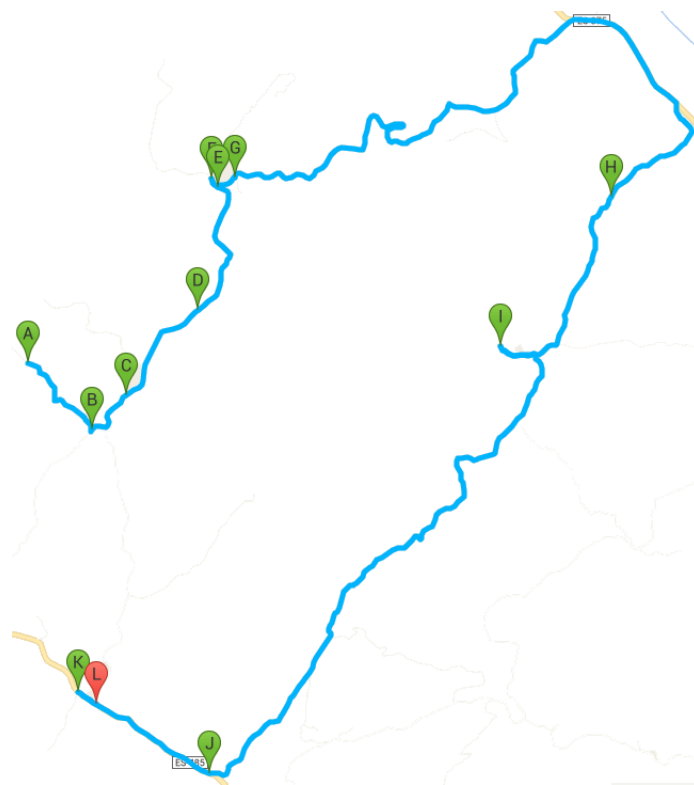


Figura 9: Viagem da Figura 8 após a remoção de ramos viários

Na figura 8, os pontos enumerados pelos literais de "A" a "L" representam pontos de embarque, enquanto que o ponto "M" representa um ponto de desembarque. Os pontos intermediários não estão ilustrados. Sobre a viagem ilustrada pela Figura 8, o algoritmo proposto por este trabalho foi aplicado para remover o máximo possível de ramos viários. O resultado é apresentado na Figura 9. É importante salientar que os literais apenas enumeram a sequência de visita a cada ponto, mas não os identificam. O ponto "E" na Figura 9 pode não ser equivalente ao ponto "E" na Figura 8.



Pode-se perceber a remoção de cinco ramos viários da Figura 8, os quais são aqueles cujos destinos são os pontos "L", "K", "E", "F" e "H". Além disso, é possível notar que o número de pontos na Figura 9 é menor do que na Figura 8, o que indica que dois pontos de embarque (no caso, "G" e "H" da Figura 8) foram remapeados para uma mesma localização (ponto "G" da Figura 9). Existem casos, entretanto, de ramos viários que não são eliminados pelo algoritmo proposto, como aquele cujo destino é o ponto "J" na Figura 8 (ou "I" na 9). Isto ocorre quando há pelo menos um estudante no ponto de grau 1 que não é capaz de se deslocar sem comprometer o limite máximo de distância permitido pela legislação.

Após a remoção dos ramos, houve uma economia, em termos de distância, de 30% para a viagem, passando de 44 km percorridos na Figura 8 para 31 km percorridos na Figura 9. É uma economia considerável, uma vez que foi obtida sobre uma rota ótima gerada previamente.

O algoritmo proposto foi executado sobre 10 soluções completas (múltiplas viagens), todas obtidas através do mesmo algoritmo de otimização anterior (de nossa autoria), para municípios do estado do Espírito Santo, Brasil. Os resultados destas execuções estão agrupados na tabela 1. Como pode ser visto na mesma tabela, o método de remoção de ramos viários foi capaz de melhorar as avaliações de soluções ótimas, previamente calculadas, em uma média de 18,6%. As soluções utilizadas somam um média de 8,9 viagens. Ainda é possível perceber que o percentual de melhoria decorrente da aplicação da metodologia proposta é proporcional à distância total percorrida nas rotas com a presença dos ramos viários. Isto faz sentido, visto que rotas curtas apresentam menor probabilidade de possuírem ramos viários e, quando presentes, menor probabilidade de que sejam longos, o que resulta em menor impacto da metodologia sobre a avaliação inicial.

Tabela 1: Tabela de resultados de aplicação do algoritmo proposto sobre soluções de roteamento de veículos escolares em municípios do estado do Espírito Santo, Brasil.

Município	Dist. sem ramos	Dist. com ramos	Percentual	Num. Viagens
São Gabriel da Palha	434,63	543,29	25%	11
São José do Calçado	109,13	118,9	9%	3
São Mateus	454,67	581,9	28%	14
São Roque do Canaã	351,58	428,9	22%	8
Serra	81,25	85,3	5%	3
Venda Nova do Imigrante	473,2	582,0	23%	15
Viana	143,29	164,7	15%	4
Vila Pavão	164,3	182,4	11%	5
Vila Valério	513,02	656,7	28%	14
Santa Leopoldina	392,76	471,3	20%	12

5. Conclusões

Um algoritmo de otimização de nossa autoria foi utilizado para gerar uma solução ótima sobre uma malha viária desenhada pelo software AutoCad Maps sobre o mapa do município de Linhares, Espírito Santo, Brasil. Esta solução era composta por apenas uma viagem, sobre a qual o algoritmo proposto foi aplicado, gerada com o objetivo de minimizar a distância total percorrida por um veículo. Com a remoção de cinco ramos viários, a viagem foi refinada em 13km em comparação a sua distância total percorrida, o que equivaleu a uma melhora de de 30% sobre sua prévia avaliação. O algoritmo proposto ainda foi aplicado sobre outras 10 soluções, desta vez uma média de 8,9 viagens (variando entre 3 e 15), e foi capaz de melhorar suas avaliações iniciais em uma média de 18,6% (variando entre 5% e 28%). Os resultados mostraram que a eficiência do método é inversamente proporcional à distância total percorrida pela solução, o que faz sentido, uma vez que soluções curtas apresentam menor probabilidade de possuírem ramos viários.



Embora os resultados apresentados por este trabalho sejam consideráveis, é importante frisar que o algoritmo proposto apresenta eficiência comprovada apenas em regiões rurais do Brasil. Devido à malha viária densa de regiões urbanas, dificilmente uma viagem neste tipo de domínio incorre no aparecimento de ramos viários em número suficiente para justificar sua remoção. Portanto, deixamos como sugestão para trabalhos futuros a adaptação desta metodologia para que seja eficiente sobre malhas mais complexas como aquelas de grandes centros urbanos.

Agradecimentos

Os autores agradecem à Universidade Federal de Ouro Preto pelo apoio financeiro a partir do Auxílio Pesquisador/PROP e às agências CAPES, FAPEMIG e CNPQ.

Referências

- Bowerman, R., Hall, B., e Calamai, P. (1995). A multi-objective optimization approach to urban school bus routing: Formulation and solution method. *Transportation Research Part A: Policy and Practice*, 29:107–123. ISSN 0965-8564.
- Chapleau, L., Ferland, J., e Rousseau, J. (1985). Clustering for routing in densely populated areas. *European Journal of Operational Research*, 20:48–57. ISSN 0377-2217.
- Defryn, C. e Sörensen, K. (2017). A fast two-level variable neighborhood search for the clustered vehicle routing problem. *Computers & Operations Research*, 83:78 – 94. ISSN 0305-0548.
- Dulac, G., Ferland, J., e Fergues, P. (1980). School bus routes generator in urban surroundings. *Computers & Operations Research*, 7:199–213. ISSN 0305-0548.
- Graham, D. e Nuttle, H. (1986). A comparison of heuristics for a school bus scheduling problem. *Transportation Research Part B: Methodological*, 20:175–182.
- Kim, B. e Jeong, S. (2009). A comparison of algorithms for origin and destination matrix generation on real road networks and an approximation approach. *Computers and Industrial Engineering*, 56:70–76. ISSN 0360-8352.
- Lawler, E. L. (1972). A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management Science*, 18:401–405.
- Li, L. e Fu, Z. (2002). The school bus routing problem: a case study. *Journal of the Operational Research Society*, p. 552–558. ISSN 0160-5682.
- Mohammed, M. A., Ghani, M. K. A., Hamed, R. I., Mostafa, S. A., Ahmad, M. S., e Ibrahim, D. A. (2017). Solving vehicle routing problem by using improved genetic algorithm for optimal solution. *Journal of Computational Science*, p. -. ISSN 1877-7503.
- Newton, R. e Thomas, W. (1969). Design of school bus routes by computer. *Socio Economic Planning Sciences*, 3:75–85. ISSN 0038-0121.
- Newton, R. e Thomas, W. (1974). Bus routing in a multi-school system. *Computers and Operations Research*, 1:213–222. ISSN 0305-0548.
- Park, J. e Kim, B. (2010). The school bus routing problem: A review. *European Journal of operational research*, 202:311–319. ISSN 0377-2217.
- Zhang, D., Cai, S., Ye, F., Si, Y.-W., e Nguyen, T. T. (2017). A hybrid algorithm for a vehicle routing problem with realistic constraints. *Information Sciences*, 394-395:167–182. ISSN 0020-0255.