

HEURÍSTICAS PARA O PROBLEMA DE AGRUPAMENTO DE ALUNOS EM ESCOLAS

Nayza Mamede Sá

Mestrado Acadêmico em Modelagem e Métodos Quantitativos, Universidade Federal do Ceará
Campus do Pici - Centro de Ciências - Bloco 910 - Fortaleza-CE, 60440-900
nayzamamede@alu.ufc.br

Bruno de Athayde Prata

Mestrado Acadêmico em Modelagem e Métodos Quantitativos, Universidade Federal do Ceará
Campus do Pici - Centro de Ciências - Bloco 910 - Fortaleza-CE, 60440-900
baprata@ufc.br

RESUMO

A alocação eficiente dos alunos às escolas mais próximas de suas residências é crucial, considerando as capacidades escolares e as demandas individuais dos alunos. Tal problema pode ser visto como um problema de agrupamento capacitado no qual as escolas possuem capacidades diferentes para cada série. Tendo em vista que esta classe de problemas é NP-difícil, a proposição de algoritmos aproximados, é de suma relevância para a resolução de instâncias de médio e grande porte. Nesta pesquisa objetiva-se a apresentação de três heurísticas construtivas para o problema de agrupamento de alunos. Experimentos computacionais em um conjunto de 120 instâncias geradas aleatoriamente apontam para a necessidade do uso de metaheurísticas para a resolução do problema, visto que o modelo proposto não conseguiu retornar soluções inteiras em um limite de tempo de 1200s.

PALAVRAS CHAVE. Problema de Agrupamento de Alunos. Otimização Combinatória. Programação Linear Inteira.

PO na Administração Pública e Educação, Otimização Combinatória, Logística e Transportes

ABSTRACT

Efficient allocation of the students to the nearest schools is crucial, taking into consideration the capacities of the schools and the demands of the students. The problem can be considered as a capacitated clustering problem, in which the schools present different capacities for each grade level. Considering that this class of problems is NP-hard, the proposition of approximate algorithms is of great importance to provide high-quality solutions within an admissible computational effort. This research aims to propose to present three constructive heuristics for the student clustering problem. Computational experiments on a set of 120 randomly generated instances indicate the need for using metaheuristics to solve the problem, as the proposed model failed to return feasible integer solutions within a time limit of 1200 seconds.

KEYWORDS. Student Clustering Problem. Combinatorial Optimization. Integer Linear Programming.

OR in the Public Administration and Education, Combinatorial Optimization, Logistics and Transportation

1. Introdução

O problema de otimização de Roteamento dos Transportes Escolares (*School Bus Routing Problem* - SBRP) é uma questão complexa que envolve a busca pela melhor maneira de alocar veículos e definir rotas eficientes para transportar os alunos de suas residências até as escolas. Esta otimização visa minimizar custos e tempo de viagem, ao mesmo tempo em que maximiza a segurança e a satisfação dos alunos.

Segundo Park e Kim [2010], o SBRP é um problema de otimização combinatória rico e prático que faz parte da rotina dos estudantes em todo o mundo. O problema foi introduzido pela primeira vez por Newton e Thomas [1969] e é formulado por meio de um conjunto de paradas, casas de estudantes, frotas de ônibus, escolas e garagens, nos quais buscamos criar rotas otimizadas visando a redução do custo operacional do serviço.

Este problema se enquadra em uma classe maior de problemas chamada de Problema de Roteamento de Veículos (VRP, do inglês *Vehicle Routing Problem*). Para Li e Fu [2002], o VRP concentra-se no uso eficiente de uma frota de veículos (por exemplo, caminhões, ônibus e carros) que devem fazer uma série de paradas para pegar e/ou entregar passageiros ou produtos. O problema exige que se especifique quais clientes devem ser entregues, por qual veículo e em que ordem, a fim de minimizar o custo total sujeito a várias restrições, como capacidade do veículo e restrições de tempo de entrega.

Nesse tipo de problema a minimização de custos é o objetivo principal, porém outros objetivos podem assumir importância, especialmente no contexto das operações de serviço no setor público. Nos estudos de Bodin e Berman [1979], sugere-se que a segurança e conveniência são dois outros objetivos nos quais certos problemas podem se concentrar. Por exemplo, no SBRP, o objetivo é minimizar o total de minutos dos alunos no ônibus, pois essa medida é percebida como altamente correlacionada com a segurança. Consequentemente, as rotas e os horários dos ônibus escolares devem ser projetados para garantir que nenhum aluno permaneça por um tempo excessivamente longo no ônibus.

Para Ellegood et al. [2020], o SBRP destaca-se como uma problemática essencial que impacta milhões de famílias ao redor do mundo. O desafio enfrentado por administradores escolares e prestadores de serviços é desenvolver e fornecer um sistema de transporte seguro, confiável e econômico para levar os alunos diariamente de casa para a escola.

De acordo com Li e Fu [2002], Ke et al. [2006] e Park e Kim [2010] o SBRP tem uma abordagem ampla e é constituído de diversos subproblemas. Para Kloeckner [2015], os subproblemas são: *Bus Stop Selection Problem* (BSSP), que consiste em selecionar as paradas dos ônibus; *Bus Selection From Available Bus Fleet* (BSFABF), onde são determinados, a partir de uma frota (homogênea ou heterogênea), os ônibus a serem utilizados; *Bus Route Generation* (BRG), onde determina-se a rota que o ônibus deve seguir; *School Bell Time Adjustment* (SBTA), que trata-se do alinhamento do tempo do transporte com o tempo de início e término da aula; *Route Scheduling* (RS), que refere-se a programação da rota e o *Student Clustering Problem* (SCP), que consiste na alocação dos alunos em escolas mais próximas quanto possível de sua residência, respeitando para isso a capacidade de aluno por série estabelecida pela escola.

Na maioria das abordagens existentes, esses subproblemas são considerados separadamente e sequencialmente. Segundo Park e Kim [2010], embora esses subproblemas não sejam independentes, mas altamente inter-relacionados, eles são tratados separadamente devido à complexidade e tamanho do problema. Conforme Ansari et al. [2021], deve-se considerar a atribuição de alunos às escolas e as gerações de rotas como um processo integrado, pois impactam significativamente a qualidade do serviço prestado aos alunos e reduzem custos operacionais.

É nesse âmbito que se estabelece o Problema de Agrupamento de Alunos (SCP), onde

objetiva-se alocar os alunos em escolas mais próximas de suas residências, respeitando algumas restrições como capacidade de cada escola e demandas de alunos por série. Segundo Batsaris et al. [2021], a alocação de estudantes é uma tarefa importante que pode afetar o funcionamento das unidades escolares e causar anomalias na alocação de recursos. Por exemplo, unidades escolares superlotadas podem exigir mais recursos (financeiros, materiais, infraestrutura e recursos humanos) para garantir o bom funcionamento e a qualidade da prestação de ensino.

De acordo com os estudos de Kloeckner [2015], o agrupamento de alunos se refere ao desafio de particionar um conjunto de alunos em agrupamentos com características semelhantes, como localização espacial e série, enquanto observa restrições relacionadas à capacidade de alocação nas escolas (*clusters*). Esse cenário é caracterizado como um Problema de Agrupamento Capacitado (*Capacitated Clustering Problem - CCP*), uma abordagem que guia a resolução do problema de agrupamento de alunos.

O Problema de Agrupamento Capacitado (*Capacitated Clustering Problem - CCP*), amplamente estudado nas últimas décadas, é um problema de otimização combinatória NP-difícil. Inicialmente, proposto por Mulvey e Beck [1984], onde define-se o CCP como a tarefa de agrupar n elementos em p *clusters* mutuamente exclusivos, onde o tamanho de cada *cluster* é limitado, objetiva-se minimizar a soma das distâncias entre cada elemento e um ponto médio do *cluster* designado. De acordo com Negreiros e Palhano [2006], para instâncias de grande escala, que são encontradas em muitas situações da vida real, para o CCP parece razoável usar abordagens heurísticas para resolver esse problema. Essas heurísticas devem ser projetadas com base no conhecimento de boas estratégias para encontrar soluções iniciais e aprimoradas.

Uma das variações do CCP é o Problema de Agrupamento Multicapitado (*Multi Capacitated Clustering Problem - MCCP*), proposto por Prata [2015], cujo objetivo é buscar agrupar elementos com características distintas em *clusters* com diferentes capacidades, minimizando a soma das distâncias entre eles. Nesse cenário, considera-se os elementos como os alunos e os *clusters* como as escolas. Para Araújo et al. [2022], a utilidade de resolver um problema MCCP é enorme tanto para fins teóricos quanto práticos, pois pode ser aplicado em diferentes situações para reduzir custos e fornecer soluções viáveis em um tempo razoável para fins de planejamento.

Este trabalho tem como objetivo apresentar um modelo de programação linear inteira e heurísticas para o problema de agrupamento de alunos em escolas. A formulação matemática e as três heurísticas construtivas propostas foram testadas em um conjunto de 120 instâncias, geradas aleatoriamente.

As seções remanescentes do artigo são descritas a seguir. Na Seção 2 é apresentada a descrição do problema em estudo, bem como a formulação matemática proposta. Na Seção 3 são apresentados métodos propostos para a solução do problema. Na Seção 5, são detalhados os experimentos computacionais conduzidos, e na Seção 6, são apresentadas as conclusões a cerca da pesquisa.

2. Descrição do Problema

O Problema de Agrupamento de Alunos (SCP) consiste na atribuição de alunos de uma série específica às escolas mais próximas de suas residências, assegurando simultaneamente que a escola selecionada tenha a capacidade necessária para alocar o aluno em sua série correspondente. A seguir, na Figura 1, é apresentada de maneira esquemática uma solução viável para o SCP.

Neste exemplo, diversos estudantes de três séries distintas, representados pelas cores amarela, lilás e verde, serão distribuídos entre três escolas. A escola 1, por exemplo, tem a capacidade máxima de receber 3 alunos lilás, 2 alunos verdes e 2 amarelos, enquanto a escola 3 pode admitir apenas um aluno do tipo amarelo e 3 do tipo lilás.

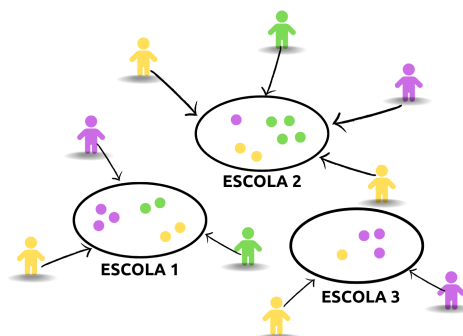


Figura 1: Solução Viável para o SCP

A notação matemática utilizada na formulação do modelo SCP, proposto por Kloeckner (2015), é apresentada a seguir.

Conjuntos:

- M : conjunto dos alunos;
- N : conjunto de escolas;
- P : conjunto de séries de ensino para alunos;

Parâmetros:

- d_{ij} : distância entre o aluno i e a escola j ;
- q_{ik} : matriz da demanda do aluno i para a série k , a qual cursa;
- w_{jk} : capacidade máxima da escola j alocar o aluno na série k ;
- f_j : custo de instalação da escola j ;
- m : cardinalidade do conjunto M ;
- n : cardinalidade do conjunto N ;
- p : cardinalidade do conjunto P .

Variáveis de decisão:

$$x_{ij} = \begin{cases} 1, & \text{se o aluno } i \text{ é alocado na escola } j; \\ 0, & \text{caso contrário.} \end{cases}$$

$$y_j = \begin{cases} 1, & \text{se a escola } j \text{ é usada;} \\ 0, & \text{caso contrário.} \end{cases}$$

O modelo matemático para o SCP é apresentado a seguir:

Função Objetivo:

$$D = \min \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_{ij} + \sum_{j=1}^n f_j y_j \quad (1)$$

Sujeito a:

$$\sum_{j=1}^n x_{ij} = 1, \quad \forall i \in M \quad (2)$$

$$\sum_{i=1}^m q_{ij} x_{ij} \leq w_{jk}, \quad \forall j \in N; \quad \forall k \in P \quad (3)$$

$$x_{ij} - y_j \leq 0 \quad \forall i \in M, \quad \forall j \in N \quad (4)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in M, \quad \forall j \in N \quad (5)$$

$$y_j \in \{0, 1\}, \quad \forall j \in N \quad (6)$$

A função objetivo (1) minimiza a distância entre as residências dos alunos i e a escola j . O primeiro termo da função refere-se ao custo de transporte do aluno i para a escola j , enquanto o segundo termo trata-se do custo fixo associado à escola j . O conjunto de restrições (2) assegura que cada aluno i seja associado a uma única escola j . Já o conjunto de restrições do tipo (3) garantem que alocação dos alunos respeitará a capacidade máxima das escolas, por série. O conjunto de restrições do tipo (4) garante que o aluno i será alocado à escola j se, e somente se, ela fizer parte da solução. Por fim, o conjunto de restrições do tipo (5) e (6) diz respeito ao caráter binário das variáveis de decisão do modelo.

Para melhor compreensão da aplicabilidade do modelo previamente exposto, procedeu-se à geração aleatória de uma instância, contemplando os dados geográficos de 20 alunos distribuídos em 5 escolas. Adicionalmente, foram consideradas as demandas de série de cada aluno e a capacidade de cada instituição escolar em alocar esses discentes em um conjunto de 4 séries do ensino fundamental II. Definiu-se, também, o custo de instalação de uma nova escola (R\$ 1.000.000,00).

Desta forma, para elucidar os dados empregados, a seguir, apresenta-se na Tabela 1 a distância euclidiana, mensurada em quilômetros, entre a residência do aluno e cada uma das escolas. Na Tabela 2 a demanda do aluno por uma série específica e na Tabela 3 é apresentada a capacidade ofertada das escolas, organizada por séries.

Tabela 1: Distância Euclidiana (Km)

Aluno	Escola 1	Escola 2	Escola 3	Escola 4	Escola 5
1	6.08	3.10	9.88	9.10	4.61
2	9.08	3.33	9.97	1.33	6.60
3	4.15	5.72	4.38	1.70	6.94
4	2.89	3.21	7.68	7.81	4.10
5	7.32	1.17	6.99	8.22	6.21
6	6.55	3.25	6.01	1.08	4.47
7	6.51	3.02	1.62	4.76	1.54
8	6.42	6.92	1.45	8.85	8.07
9	8.25	4.56	9.25	1.53	3.71
10	5.83	5.11	9.07	3.11	5.63
11	9.22	2.01	5.46	1.22	2.92
12	8.78	8.31	2.34	9.06	7.48
13	2.75	5.45	4.28	7.21	4.65
14	8.70	8.97	8.57	7.74	9.31
15	7.66	4.83	8.58	6.01	2.95
16	1.94	5.38	1.56	1.80	6.96
17	5.05	5.36	3.59	5.45	6.50
18	4.55	2.79	2.59	2.02	4.10
19	5.84	5.16	6.64	9.02	1.98
20	8.40	9.11	3.36	7.68	5.18

Fonte: Elaborada pelos autores

Tabela 2: Demanda de alunos por série

Aluno	6º ano	7º ano	8º ano	9º ano
1	-	-	-	X
2	-	-	X	-
3	-	-	-	X
4	-	-	-	X
5	-	X	-	-
6	-	-	X	-
7	X	-	-	-
8	X	-	-	-
9	-	-	-	X
10	-	-	X	-
11	X	-	-	-
12	-	-	-	X
13	-	-	-	X
14	X	-	-	-
15	X	-	-	-
16	-	-	-	X
17	-	X	-	-
18	-	X	-	-
19	-	X	-	-
20	-	X	-	-

Fonte: Elaborada pelos autores

Tabela 3: Capacidade das Escolas por Série

	6º ano	7º ano	8º ano	9º ano
Escola 1	3	1	3	3
Escola 2	0	2	0	1
Escola 3	4	4	4	2
Escola 4	4	1	2	0
Escola 5	1	1	3	1

Fonte: Elaborada pelos autores

O modelo foi implementado na linguagem *Python* e utilizado o solver CPLEX, consolidando a alocação dos alunos nas escolas, respeitando a capacidade de cada instituição de ensino, correspondente à sua série específica. Na Tabela 4 é apresentado o resultado da alocação dos estudantes, respeitando a demanda específica de cada aluno. Adicionalmente, na Tabela 5, são destacadas as escolas que fazem parte da solução e na Tabela 6 as capacidades efetivamente utilizadas, demonstrando que a oferta de cada escola foi atendida.

Tabela 4: Alocação de Alunos nas Escolas

Aluno	Série	Escola 1	Escola 2	Escola 3	Escola 4	Escola 5
1	9º ano	-	X	-	-	-
2	8º ano	-	-	-	-	X
3	9º ano	-	-	X	-	-
4	9º ano	X	-	-	-	-
5	7º ano	-	X	-	-	-
6	8º ano	-	-	-	-	X
7	6º ano	-	-	X	-	-
8	6º ano	-	-	X	-	-
9	9º ano	-	-	-	-	X
10	8º ano	X	-	-	-	-
11	6º ano	-	-	X	-	-
12	9º ano	-	-	X	-	-
13	9º ano	X	-	-	-	-
14	6º ano	-	-	X	-	-
15	6º ano	-	-	-	-	X
16	9º ano	X	-	-	-	-
17	7º ano	-	-	X	-	-
18	7º ano	-	X	-	-	-
19	7º ano	-	-	-	-	X
20	7º ano	-	-	X	-	-

Fonte: Elaborada pelos autores

Tabela 5: Escolas Participantes da Solução

Escola 1	Escola 2	Escola 3	Escola 4	Escola 5
X	X	X	-	X

Fonte: Elaborada pelos autores

Tabela 6: Capacidades utilizadas

	6º ano	7º ano	8º ano	9º ano
Escola 1	0	0	1	3
Escola 2	0	2	0	1
Escola 3	4	2	0	2
Escola 4	0	0	0	0
Escola 5	1	1	2	1

Fonte: Elaborada pelos autores

3. Métodos Propostos

A estratégia inicial para resolver o modelo será utilizando o solver de otimização CPLEX; no entanto, é imperativo reconhecer que o SCP é considerado um problema NP-difícil, visto que se considerarmos que as escolas possuem apenas um tipo de série, ele torna-se um caso particular do Problema de Agrupamento Capacitado (CCP), o qual é NP-difícil [Mulvey e Beck, 1984]. Assim, é plausível considerar que o CPLEX não seja capaz de encontrar uma solução dentro do limite de tempo estabelecido, dependendo das dimensões das instâncias envolvidas. Em vista disso, uma alternativa proposta consiste na implementação de Heurísticas.

No intuito de encontrar soluções viáveis, para instâncias de grande porte, propõe-se três Heurísticas construtivas gulosas descritas a seguir:

A Heurística 1 (Algoritmo 1) consiste nos seguintes passos: Para cada escola $j \in M$, o algoritmo itera sobre as séries k . Para cada série, os alunos são selecionados com base na demanda específica da série e são ordenados de acordo com sua proximidade geográfica com as escolas. A alocação dos alunos é então realizada, atribuindo o aluno i mais próximo da escola j caso tenha capacidade disponível. Após a alocação de um aluno, a capacidade da escola é atualizada e o aluno é removido do conjunto de alunos não alocados.

Na Heurística 2 (Algoritmo 2), para cada série k , filtra-se as escolas com capacidade disponível. Para cada escola j filtrada, os alunos são selecionados de acordo com a demanda específica dessa série e ordenados com base na proximidade da escola. A alocação dos alunos é feita de maneira sequencial, começando pelas escolas com maior capacidade, até que a capacidade de cada escola seja preenchida.

Já na Heurística 3 (Algoritmo 3), para cada aluno $i \in M$ não alocado, o algoritmo identifica a série do aluno e filtra as escolas que possuem capacidade para essa série. Em seguida, aloca o aluno na escola mais próxima. O algoritmo continua esse processo até que todos os alunos sejam alocados.

Algoritmo 1 Heurística 1

```

1: Entrada ( $M, N, P, m, n, p, d, w, q, f$ )
2:    $x \leftarrow \text{zeros}(m, n)$ 
3:    $y \leftarrow \text{zeros}(n)$ 
4:    $\text{custo} \leftarrow 0$ 
5:   Enquanto  $M$  não estiver vazio faça
6:     Para cada escola  $j \in N$  faça
7:       Para cada série  $k \in P$  faça
8:         Se  $w[j][k] > 0$  então
9:            $\text{alunos\_filtrados} \leftarrow$  lista de alunos  $i \in M$  tal que  $q[i][k] = 1$ 
10:           $\tilde{i} \leftarrow$  aluno  $i \in \text{alunos\_filtrados}$  com  $\min\{d[i][j]\}$ 
11:          Se  $\tilde{i}$  não é nulo então
12:            Aloca o aluno  $\tilde{i}$  na escola  $j$ :  $x[\tilde{i}][j] \leftarrow 1$ 
13:            Remove o aluno  $\tilde{i}$  de  $M$ 
14:            Atualiza a capacidade da escola  $j$ :  $w[j][k] \leftarrow w[j][k] - 1$ 
15:            Calcula o custo da alocação:  $\text{custo} \leftarrow \text{custo} + d[\tilde{i}][j]$ 
16:          fim Se
17:        fim Se
18:      fim Para
19:    fim Para
20:    Se  $\sum_{i=1}^m x[i][j] > 0$  então
21:       $y[j] \leftarrow 1$ 
22:       $\text{custo} \leftarrow \text{custo} + y[j] \times f$ 
23:    fim Se
24:  retorne  $x, y, \text{custo}$ 
  
```

Algoritmo 2 Heurística 2

```

1: Entrada ( $M, N, P, m, n, p, d, w, q, f$ )
2:    $x \leftarrow \text{zeros}(m, n)$ 
3:    $y \leftarrow \text{zeros}(n)$ 
4:    $\text{custo} \leftarrow 0$ 
5:   Enquanto  $M$  não estiver vazio faça
6:     Para cada série  $k$  faça
7:       Se  $w[j][k] > 0$  então
8:          $\text{alunos\_filtrados} \leftarrow$  lista de alunos  $i \in M$  onde  $q[i][k] = 1$ 
9:         Ordena os alunos filtrados pela distância  $d_{ij}$ 
10:        Para cada aluno  $i$  em  $\text{alunos\_filtrados}$  faça
11:          Se  $w[j][k] > 0$  então
12:            Aloca o aluno  $i$  na escola  $j$ 
13:            Remove o aluno  $i$  de  $M$ 
14:            Atualiza a capacidade da escola  $j$ :  $w[j][k] \leftarrow w[j][k] - 1$ 
15:            Calcula o custo da alocação:  $\text{custo} \leftarrow \text{custo} + d[i][j]$ 
16:            Se  $w[j][k] = 0$  então
17:              pare
18:            fim Se
19:          fim Se
20:        fim Para
21:      fim Se
22:    fim Para
23:    Se  $\sum_{i=1}^m x[i][j] > 0$  então
24:       $y[j] \leftarrow 1$ 
25:       $\text{custo} \leftarrow \text{custo} + y[j] \times f$ 
26:    fim Se
27:  retorne  $x, y, \text{custo}$ 
    
```

Algoritmo 3 Heurística 3

```

1: Entrada ( $M, N, P, m, n, p, d, w, q, f$ )
2:    $x \leftarrow \text{zeros}(m, n)$ 
3:    $y \leftarrow \text{zeros}(n)$ 
4:    $\text{custo} \leftarrow 0$ 
5:   Enquanto  $M$  não estiver vazio faça
6:     Para cada aluno  $i$  em  $M$  faça
7:       Para cada série  $k$  de 1 até  $P$  faça
8:         Se  $q[i][k] = 1$  então
9:            $\text{escolas\_filtradas} \leftarrow$  lista de escolas  $j \in N$  onde  $w[j][k] > 0$ 
10:           $\bar{j} \leftarrow \min\{d[i][j] : j \in \text{escolas\_filtradas}\}$ 
11:          Aloque o aluno  $i$  na escola mais próxima  $\bar{j}$ 
12:          Remova o aluno  $i$  de  $M$ 
13:          Atualize a capacidade da escola  $\bar{j}$ :  $w[\bar{j}][k] \leftarrow w[\bar{j}][k] - 1$ 
14:          Calcule o custo da alocação:  $\text{custo} \leftarrow \text{custo} + d[i][\bar{j}]$ 
15:          Se  $y[\bar{j}] = 0$  então
16:            Atualize  $y[\bar{j}]$  para 1
17:          fim Se
18:        pare
19:      fim Se
20:    fim Para
21:  fim Para
22:  fim Enquanto
23:   $\text{custo} \leftarrow \text{custo} + \sum_{j \in N} y[j] \times f$ 
24:  retorne  $x, y, \text{custo}$ 
    
```

4. Experimentos Computacionais

Instâncias de teste para o SCP não estão disponíveis na literatura, portanto, foram elaboradas instâncias com os seguintes valores: $m \in \{100, 250, 500, 1000\}$, $n \in \{\frac{m}{5}, \frac{3m}{10}\}$ e $p \in \{6, 7, 8\}$. Para cada combinação de valores, foram geradas 5 instâncias distintas, totalizando 120 instancias. A Tabela 7 apresenta as características de cada instância, que foram organizadas em oito grupos para melhor compreensão e análise. Cada grupo representa um conjunto de instâncias que possuem

características ou propriedades semelhantes.

Tabela 7: Instâncias

INSTÂNCIA	m	n	p	INSTÂNCIA	m	n	p	INSTÂNCIA	m	n	p	INSTÂNCIA	m	n	p
SCP_1	100	20	6	SCP_31	250	50	6	SCP_61	500	100	6	SCP_91	1000	200	6
SCP_2	100	20	6	SCP_32	250	50	6	SCP_62	500	100	6	SCP_92	1000	200	6
SCP_3	100	20	6	SCP_33	250	50	6	SCP_63	500	100	6	SCP_93	1000	200	6
SCP_4	100	20	6	SCP_34	250	50	6	SCP_64	500	100	6	SCP_94	1000	200	6
SCP_5	100	20	6	SCP_35	250	50	6	SCP_65	500	100	6	SCP_95	1000	200	6
SCP_6	100	20	7	SCP_36	250	50	7	SCP_66	500	100	7	SCP_96	1000	200	7
SCP_7	100	20	7	SCP_37	250	50	7	SCP_67	500	100	7	SCP_97	1000	200	7
SCP_8	100	20	7	SCP_38	250	50	7	SCP_68	500	100	7	SCP_98	1000	200	7
SCP_9	100	20	7	SCP_39	250	50	7	SCP_69	500	100	7	SCP_99	1000	200	7
SCP_10	100	20	7	SCP_40	250	50	7	SCP_70	500	100	7	SCP_100	1000	200	7
SCP_11	100	20	8	SCP_41	250	50	8	SCP_71	500	100	8	SCP_101	1000	200	8
SCP_12	100	20	8	SCP_42	250	50	8	SCP_72	500	100	8	SCP_102	1000	200	8
SCP_13	100	20	8	SCP_43	250	50	8	SCP_73	500	100	8	SCP_103	1000	200	8
SCP_14	100	20	8	SCP_44	250	50	8	SCP_74	500	100	8	SCP_104	1000	200	8
SCP_15	100	20	8	SCP_45	250	50	8	SCP_75	500	100	8	SCP_105	1000	200	8
SCP_16	100	30	6	SCP_46	250	75	6	SCP_76	500	150	6	SCP_106	1000	300	6
SCP_17	100	30	6	SCP_47	250	75	6	SCP_77	500	150	6	SCP_107	1000	300	6
SCP_18	100	30	6	SCP_48	150	75	6	SCP_78	500	150	6	SCP_108	1000	300	6
SCP_19	100	30	6	SCP_49	150	75	6	SCP_79	500	150	6	SCP_109	1000	300	6
SCP_20	100	30	6	SCP_50	150	75	6	SCP_80	500	150	6	SCP_110	1000	300	6
SCP_21	100	30	7	SCP_51	150	75	7	SCP_81	500	150	7	SCP_111	1000	300	7
SCP_22	100	30	7	SCP_52	150	75	7	SCP_82	500	150	7	SCP_112	1000	300	7
SCP_23	100	30	7	SCP_53	150	75	7	SCP_83	500	150	7	SCP_113	1000	300	7
SCP_24	100	30	7	SCP_54	250	75	7	SCP_84	500	150	7	SCP_114	1000	300	7
SCP_25	100	30	7	SCP_55	250	75	7	SCP_85	500	150	7	SCP_115	1000	300	7
SCP_26	100	30	8	SCP_56	250	75	8	SCP_86	500	150	8	SCP_116	1000	300	8
SCP_27	100	30	8	SCP_57	250	75	8	SCP_87	500	150	8	SCP_117	1000	300	8
SCP_28	100	30	8	SCP_58	250	75	8	SCP_88	500	150	8	SCP_118	1000	300	8
SCP_29	100	30	8	SCP_59	250	75	8	SCP_89	500	150	8	SCP_119	1000	300	8
SCP_30	100	30	8	SCP_60	250	75	8	SCP_90	500	150	8	SCP_120	1000	300	8

Fonte: Elaborada pelos autores

As matrizes de distância e capacidade foram geradas aleatoriamente seguindo uma distribuição uniforme $U[1, 100]$ e $U[1, 5]$, respectivamente. A matriz de demanda é uma matriz binária onde cada aluno tem demanda para uma única serie.

As instâncias criadas estão hospedadas na plataforma *GitHub* e poderão ser acessadas a partir do link: <https://github.com/NayzaMamede/SCP.git>. Este repositório contém tanto as instâncias quanto o modelo de programação inteira, implementado em *Python*, para resolver o SCP.

Realizou-se a experiência computacional em um PC com processador Intel(R) Core(TM) i5-4210U CPU 1.70GHz e 8 GB de memória e sistema operacional Windows 10 PRO. Para o modelo MILP, o resolutor usado foi o IBM ILOG CPLEX versão 22.1.1. A implementação tanto do modelo quanto das heurísticas foi realizada utilizando a linguagem *Python*. Para a modelagem, empregou-se a biblioteca *Pyomo*, enquanto para as heurísticas utilizou-se a biblioteca *NumPy*. Para fins de comparação, foram estabelecidos tempos limites para a execução dos métodos propostos para todas as instâncias geradas: 300, 600, 900 e 1200 segundos.

Na Tabela 8, são apresentados os resultados dos testes computacionais realizados com o solver CPLEX. São exibidos os tempos médios de execução e a quantidade de soluções encontradas para cada grupo de instâncias, organizados por limites de tempo predefinidos. Para uma compreensão mais clara, os gráficos correspondentes, podem ser encontrados na Figura 2 e na Figura 3.

É possível observar que, nos grupos de instâncias 5 e 6, em que $m = 500$, o resolutor retornou menos de 20% de soluções ótimas, sem haver avanços significativos com o aumento do limite de tempo. Já nos grupos 7 e 8, com $m = 1000$, o resolutor não encontrou soluções ótimas. Conclui-se, portanto, que em um contexto real, como um sistema educacional de um município, por exemplo, com uma quantidade substancial de alunos e escolas, o MILP pode não ser eficiente para lidar com o SCP.

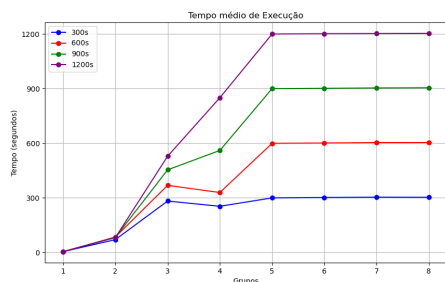
Os resultados dos experimentos realizados com as heurísticas implementadas, são apre-

Tabela 8: Resultados CPLEX

	300 segundos		600 segundos		900 segundos		1200 segundos	
Grupo	Tempo(s)	Soluções	Tempo(s)	Soluções	Tempo(s)	Soluções	Tempo(s)	Soluções
1	4,58	15	4,13	15	5,08	15	5,49	15
2	66,48	14	98,73	14	115,63	14	133,45	14
3	255,84	4	429,45	10	499,02	11	560,93	13
4	238,20	5	419,04	7	558,20	9	704,54	9
5	286,86	1	567,04	2	827,99	2	1079,07	2
6	300,00	0	600,00	0	886,18	1	1165,63	1
7	300,00	0	600,00	0	900,00	0	1200,00	0
8	300,00	0	600,00	0	900,00	0	1200,00	0

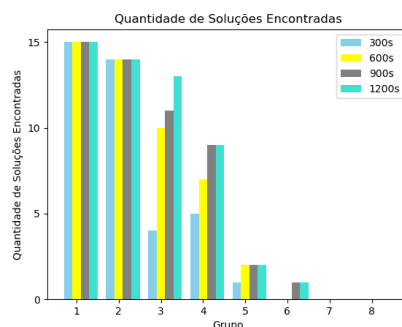
Fonte: Elaborada pelos autores

Figura 2: Tempo Médio (CPLEX)



Fonte: Elaborada pelos autores

Figura 3: Quantidade de Soluções



Fonte: Elaborada pelos autores

sentados a seguir. Para fins de comparação, utiliza-se o indicador Desvio Percentual Relativo (DPR), que compara as soluções encontradas pelo CPLEX no maior limite de tempo estabelecido (1200 segundos) com os resultados obtidos pelas Heurísticas, a seguir a expressão para o cálculo do DPR:

$$DPR_{ih} = \frac{\text{Solução_Heurística}_{ih} - \text{Solução_CPLEX}_{ih}}{\text{Solução_CPLEX}_{ih}}, \quad (7)$$

onde i representa as instâncias geradas, $i = 1...120$ e h as Heurísticas implementadas, $h = 1, 2, 3$.

Na Tabela 9 é apresentado o tempo médio de execução das heurísticas e o DPR médio em relação ao CPLEX. Nas Figuras 4, 5 e 6 são ilustrados os boxplots do DPR médio das Heurísticas 1, 2 e 3, respectivamente. Na Figura 7, um gráfico de dispersão compara o DPR das Heurísticas.

5. Conclusões

Este estudo dedica-se ao problema de agrupamento de alunos em escolas, com o intuito de minimizar as distâncias geográficas entre suas residências e as instituições de ensino, enquanto atende às demandas individuais dos alunos e respeita as capacidades das escolas em cada série. Para abordar esse problema, desenvolveu-se um modelo de programação linear inteira e três heurísticas construtivas de forma a realizar a alocação de alunos às escolas de forma eficiente.

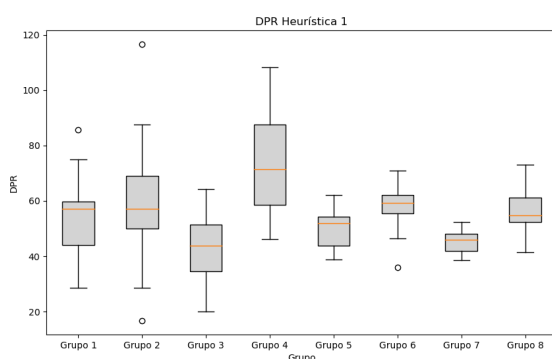
Os experimentos computacionais, realizados em um conjunto de 120 instâncias geradas aleatoriamente, destacaram a necessidade do uso de heurísticas e metaheurísticas para a resolução do problema, devido à sua natureza NP-difícil. Enquanto o modelo proposto não conseguiu retornar soluções inteiras viáveis dentro de um limite de tempo razoável, as heurísticas construtivas apresentaram resultados em tempo computacionais admissíveis, retornando soluções viáveis para todas as instâncias.

Tabela 9: Resultados Computacionais Heurísticas

	Heurística 1		Heurística 2		Heurística 3	
Grupo	Tempo (s)	DPR (%)	Tempo (s)	DPR (%)	Tempo (s)	DPR (%)
1	0,0021	53,75	0,0012	113,09	0,0042	176,99
2	0,0022	60,16	0,0063	165,42	0,0031	324,67
3	0,0144	42,61	0,0200	145,2	0,0127	213,78
4	0,0211	74,29	0,0246	178,25	0,0195	394,85
5	0,0869	50,45	0,0809	155,91	0,0917	246,92
6	0,0596	58,75	0,0997	182,09	0,1464	420,54
7	0,2534	45,54	0,2823	165,37	0,3192	252,81
8	0,1719	56,56	0,4133	186,12	0,2632	401,14

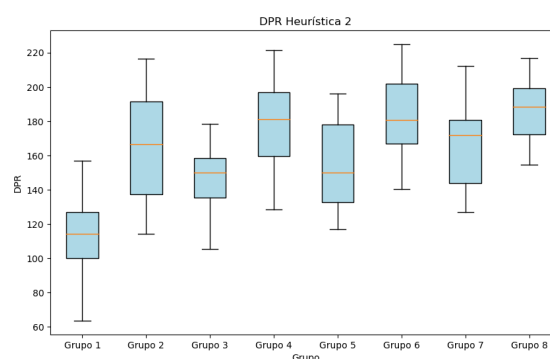
Fonte: Elaborada pelos autores

Figura 4: DPR médio: Heurística 1



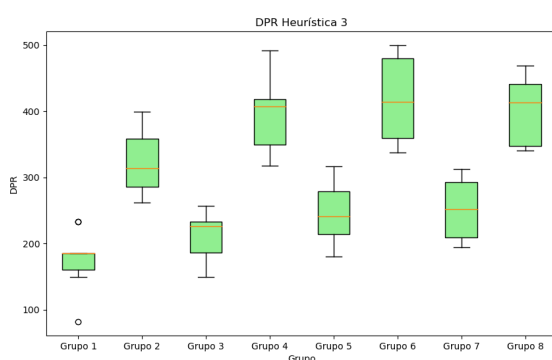
Fonte: Elaborada pelos autores

Figura 5: DPR médio: Heurística 2



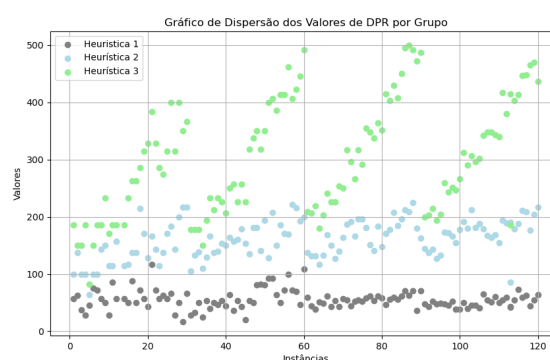
Fonte: Elaborada pelos autores

Figura 6: DPR médio: Heurística 3



Fonte: Elaborada pelos autores

Figura 7: Comparativo de DPR



Fonte: Elaborada pelos autores

Como parte dos experimentos, conduziu-se uma análise comparativa entre as heurísticas propostas e o solver CPLEX, utilizando o DPR como métrica de desempenho. Os resultados revelam que a Heurística 1 alcançou os menores valores de DPR em comparação com as demais. No entanto, observamos que ainda existe uma distância considerável em relação às melhores soluções encontradas pelo CPLEX. Isso indica a necessidade de explorar metaheurísticas ou outros méto-

dos mais robustos para aprimorar ainda mais os resultados e atender aos requisitos de desempenho desejados.

Como sugestões para trabalhos futuros (ora em desenvolvimento pelos autores), podem ser destacadas a geração de novos conjuntos de dados (instâncias) para o problema em estudo, bem como a proposição de novos métodos de resolução, tais como metaheurísticas.

6. Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, da Secretaria Municipal de Educação (SME) de Fortaleza, e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processos 407151/2021-4 e 309755/2021-2.

Referências

- Ansari, A., Farrokhvar, L., e Kamali, B. (2021). Integrated student to school assignment and school bus routing problem for special needs students. *Transportation Research Part E: Logistics and Transportation Review*, 152:102416.
- Araújo, K. A. G. d., Guedes, J. B., e de Athayde Prata, B. (2022). Hybrid matheuristics for the multi-capacitated clustering problem. *RAIRO-Operations Research*, 56(3):1167–1185.
- Batsaris, M., Kavrouidakis, D., Hatjiparaskevas, E., e Agourogianis, P. (2021). Spatial decision support system for efficient school location-allocation. *European Journal of Geography*, 12(4).
- Bodin, L. D. e Berman, L. (1979). Routing and scheduling of school buses by computer. *Transportation Science*, 13(2):113–129.
- Ellegood, W. A., Solomon, S., North, J., e Campbell, J. F. (2020). School bus routing problem: Contemporary trends and research directions. *Omega*, 95:102056.
- Ke, X., Aneja, Y., e Caron, R. (2006). The school bus routing and scheduling problem with heterogeneous bus capacity: Formulations and their solutions.
- KloECKner, N. V. d. R. (2015). O problema de nucleação de alunos no transporte escolar.
- Li, L. Y. e Fu, Z. (2002). The school bus routing problem: a case study. *Journal of the Operational Research Society*, 53:552–558.
- Mulvey, J. M. e Beck, M. P. (1984). Solving capacitated clustering problems. *European Journal of Operational Research*, 18(3):339–348.
- Negreiros, M. e Palhano, A. (2006). The capacitated centred clustering problem. *Computers & operations research*, 33(6):1639–1663.
- Newton, R. M. e Thomas, W. H. (1969). Design of school bus routes by computer. *Socio-Economic Planning Sciences*, 3(1):75–85.
- Park, J. e Kim, B.-I. (2010). The school bus routing problem: A review. *European Journal of Operational Research*, 202(2):311–319. ISSN 0377-2217.
- Prata, B. (2015). The multi capacitated clustering problem. Technical report, Technical report, Universidade Federal do Ceará.