COMPARATIVE ANALYSIS OF THE RESULTS OF THE GENETIC ALGORITHM METAHEURISTIC AND THE SLIME MOLD ALGORITHM METAHEURISTIC IN SCHOOL TRANSPORTATION ROUTING OPTIMIZATION

Bruno Reis Moreira Nacano
Instituto Tecnológico de Aeronáutica
Pça. Mal. do Ar Eduardo Gomes 50 (Vila das Acácias), São José dos Campos, SP, 12228-900
Universidade Federal de São Paulo
Rua Talim 330, São José dos Campos, SP, 12231-280
bruno.nacano.101303@ga.ita.br

Jefferson Mika
Instituto Tecnológico de Aeronáutica
Pça. Mal. do Ar Eduardo Gomes 50 (Vila das Acácias), São José dos Campos, SP, 12228-900
Universidade Federal de São Paulo
Rua Talim 330, São José dos Campos, SP, 12231-280
jefferson.mika.101423@ga.ita.br

Jonathan Martinez Tsen
Instituto Tecnológico de Aeronáutica
Pça. Mal. do Ar Eduardo Gomes 50 (Vila das Acácias), São José dos Campos, SP, 12228-900
Universidade Federal de São Paulo
Rua Talim 330, São José dos Campos, SP, 12231-280
jonathan.tsen.101245@ga.ita.br

ABSTRACT

The objective of this work was to compare the recent Metaheuristic Slime Mold Algorithm (SMA) with Methaurisc Genetica (AG) in solving the school bus routing problem. A variation of the traveling salesman's discrete optimization problem was chosen to find the best route for the vehicle lines of a school unit in Caraguatatuba-SP. To obtain its ideal solution, the problem was formulated as a vehicle routing problem and modeled in the AMPL language, solved on the NEOS server using the Gurobi solver. Even though AG had the best, SMA managed to solve the problem, presenting viable routes for the work in question. No works were found related to the routing of vehicles, buses or school buses using the SMA algorithm, with the possibility that this work is the first related to this use.

KEYWORDS. School Bus Transportation, Slime Mould Algorithm, Operations Research.

MH - Metaheuristics, L&T - Logistic and Transportation

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

RESUMO

O objetivo deste trabalho foi comparar a recente Metaheurística Slime Mold Algorithm (SMA) com a Metaheurística Genética (AG) na solução do problema de roteamento de ônibus escolares. Foi escolhida uma variação do problema de otimização discretamente do caixeiro viajante para encontrar a melhor rota das linhas de veículo de uma unidade escolar de Caraguatatuba-SP. Para obter sua solução ideal, o problema foi formulado como um problema de roteamento de veículo e modelado na linguagem AMPL, resolvido no servidor NEOS utilizando o solucionador Gurobi. Mesmo a AG tendo os melhores, a SMA conseguiu solucionar o problema, apresentando rotas viáveis para o trabalho em questão. Não foram encontrados trabalhos relacionados ao roteamento de veículos, ônibus ou ônibus escolares com a utilização do algoritmo SMA, havendo a possibilidade desse trabalho ser o primeiro relacionado a essa utilização.

PALAVRAS CHAVES. Transporte escolar. Slime Mould Algorithm. Pesquisa Operacional.

MH - Meta-heurísticas, L&T - Logística e Transportes

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

1. Introduction

Access to education in Brazil is a multifaceted challenge that demands multidisciplinary efforts to ensure the continuous improvement of education in the country. Within this context, one of the problems faced is ensuring the transportation of students from their homes to educational centers. Particularly when considering Brazil's considerable geographical dimensions, the transportation problem takes on proportions of high cost for public funds and, consequently, for the entire population. Therefore, in the area of Operations Research, this work proposes to evaluate a method for optimizing school transportation routes in order to optimize available resources and assist in the dimensioning of the vehicle fleet for school units.

This work proposes evaluating a recent metaheuristic called Slime Mould Algorithm (SMA) [Li S et al 2020]. As its name suggests, this metaheuristic emerged based on the behavior of the commonly called "slime mold," however, without taxonomic meaning, i.e., it is not actually a fungus in taxonomic characterization. This particular organism, Physarum polycephalum, is characterized by a slimy mass that exhibits an expansive, floating behavior and tends to spread over surfaces, creating networks of connections between distinct food sources. This idea presents new resources with a mathematical model that provides an ideal path to connect elements and has a high exploration capacity.

The evaluation will be applied to a specific case interested in optimizing the school transportation routes of the Associação de Pais e Amigos dos Excepcionais (Apae) in Caraguatatuba. In this proposed problem, transportation is intended to pick up students from their homes, hence the need to optimize the fleet size to pick up each student at their respective residence.

In summary, the objective of this work is to solve a Discrete Optimization Problem to find the best route for the school bus lines of this school unit in Caraguatatuba. The availability of 20 points will be simulated, divided into n lines that vary according to the capacity of the vehicles available in the fleet, representing the transportation demand for a period of the day. This combination aims to minimize the value of the objective function under the constraints established by the distance criteria between the points on the line. To obtain its optimal solution, the problem was formulated as a model in the AMPL language, using the Gurobi solver, in order to obtain its optimal solution. A Genetic Algorithm (GA) metaheuristic model was also used as a way to compare the results for SMA evaluation.

2. Routing Problem

The Vehicle Routing Problem (VRP) plays a critical and fundamental role in the success of the logistics and transportation process [Giaglis et al. 2004]. Its essence is to serve a predefined number of customers using a fleet of vehicles, seeking the route that minimizes the total cost, therefore, it can be modeled as a combinatorial optimization problem. The problem has been the subject of study in the field of Computing for over half a century [Dantzig and Ramser 1959]. It represents a variation of the Traveling Salesman Problem [Gutin and Punnen 2002, Kanda 2014], classified as NP-hard [Lenstra and Rinnooy Kan 1981].

The problem will be formulated as a model known as the “Traveling Salesman Problem” (TSP), one of the most famous and studied problems in the field of combinatorial optimization. The TSP aims to determine the shortest path a traveling salesman can take, visiting each city exactly once and returning to the city of origin [KARP 1975]. In this sense, the problem studied can be modeled as the Multiple Traveling Salesman Problem (mTSP), a generalization of the traveling salesman problem, where more than one traveler is allowed. That is, there are multiple vehicles to pick up students leaving the same school.

3. Related Works

This section presents a synthesis of works that explore traditional solutions for vehicle routing and their application in school transportation, also known as the School Transportation Problem (STP). In addition, the respective methods applied to the solution of this type of problem are explained, as well as the qualitative results pertinent to each methodology.

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

The work of Porto et al. (2018) presents a comparison between different routing tools applied to STP in rural areas. In this context, the group of researchers evaluates a program based on Geographic Information Systems (GIS) in order to reduce transportation service costs, which in turn are compared against a well-known commercial program called ArcGIS. This GIS methodology relates spatial data (streets, map addresses, etc.) with tabular data (example: GPS coordinates), thus, it is possible to manipulate the information of each stop on a map, therefore, a more focused way to solve the routing problem in different situations, in this case, school transportation.

The work of Caldas (2022) applies an Iterated Local Search (ILS) metaheuristic to serve 13,664 students in rural areas of the state of Rio de Janeiro, Brazil. To achieve this objective, the School Bus Routing Problem was considered, involving a heterogeneous fleet, to minimize the overall cost, taking into account vehicle capacity constraints and maximum travel distance. The method was applied to data from the State of Rio de Janeiro, aiming to bridge the gap between school practice, transportation practices and academic models, as well as to quantify the economic potential and benefits. Computational experiments demonstrated that, when comparing the results of the method with the routes currently used, an average cost reduction of 40.5% and a reduction in average mileage per student of 46.0% are obtained [Caldas, 2022].

Melo (2021) presents a study that addressed an extension of the School Bus Routing Problem (SBRP), which encompasses subproblems such as locating bus stops, assigning students to selected stops, and routing buses for multiple periods within a single school. The objective was to minimize the total distance traveled by the fleet in the analyzed periods, considering the capacity of the vehicles and the maximum walking distance of the students to the stops. A mathematical model for the problem was presented, and an algorithm based on ILS and Variable Neighborhood Descent (VND) metaheuristics was proposed. In addition, heuristics for inserting and removing stops considering multiple periods were developed, known as Iterated Local Search with random variable neighborhood descent and removal (ILS-RVND-R). The proposed algorithm was run for a given number of SBRP instances, demonstrating favorable performance and low computational cost for the problems. [Melo 2021]

A case study focused on the school bus routing problem was described by Li and Fu (2002) and formulated as a multi-objective combinatorial optimization problem. The objectives considered included minimizing the total number of buses required, the total travel time spent by students to collection points (a crucial factor for schools and parents of students) and the total travel time in the transport vehicle. An efficient heuristic algorithm was proposed and executed, producing numerical results that demonstrated efficacy with a considerable reduction in total travel time compared to baseline practices [Li and Fu 2002].

Finally, related to the algorithm under study, the Slime Mold Algorithm is inspired by Physarum polycephalum, this organism has been analyzed as an ideal biological substrate for transportation networks. When exposed to multiple nutrient sources, the slime mold spreads, colonizes the sources and connects them through a network of protoplasmic tubes, supposedly optimized for the transfer of nutrients and metabolites. The formation of the slime mold protoplasmic network resembles the development of human transportation systems. When the main urban areas of China were represented by oat flakes, and the slime mold was inoculated in Beijing, the slime mold colonized all urban areas revealing that P. polycephalum offers a very effective match for Chinese highway networks [Adamatzky et al., 2013]. It has been demonstrated that the Slime Mold Physarum polycephalum forms networks efficiently, being used in the Tokyo railway system. It is concluded that the central mechanisms necessary for the formation of adaptive networks can be captured in a biologically inspired mathematical model that can be useful to guide the construction of networks in other domains [Tero et al, 2010]. An algorithm, based on the behavior of Slime Mold, mapped the location of 37,000 galaxies present in the universe. It was possible to reproduce a network of the cosmos that possibly links the galaxies [Burchett et al, 2020].

In summary, different route optimization problems for the school problem were presented. Depending on the problem formulation, it can be seen that as the complexity of the model increases, more robust solution methods are needed, which justifies the use of metaheuristics to solve a problem that essentially resembles the classic traveling salesman.

4. GA and SMA Metaheuristics

Two Metaheuristics will be used for comparison of results. One will be the Genetic Algorithm (GA), created by John Holland in 1975, based on Charles Darwin's theory of natural evolution. Adapting the

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

traveling salesman problem to the GA following the logic proposed in Figure 1 by Chen et al. [Chen et al. 2010], the coordinates of the points of the problem will be determined, each point being represented by a pair of longitude and latitude coordinates. Distances between points were calculated using a Euclidean function, which calculates the Euclidean distance between two points based on their coordinates. In each generation, the population will be ranked based on fitness, then a new population will be created through crossover and mutation. Thus, elitism is implemented, where the best individual from the previous generation is preserved in the new population. The algorithm continues until the results stabilize after a set number of generations. The best solution found is the route with the shortest distance traveled [Chen et al. 2010].

GA Algorithm

```
1:   T = O;
2:   Generate initial population P(0);
3:   for all each individual i o factual population P(t) do
4:       Evaluate fitness of individual i;
5:   end for
6:   while stopping criterion is not satisfied do
7:       t = t + 1;
8:       Select population P(t) from P(t-1);
9:       Apply cross operator on P(t);
10:      Apply mutation operator on P(t);
11:      Evaluate P(t);
12:  end while
```

Figura 1: Pseudo-algoritmo AG, Chen et al. 2010.

In turn, the SMA algorithm shown in Figure 2 represents a meta-heuristic approach recently presented by Shimin Li [Li et al. 2020].

Algorithm 1 Pseudo-code of SMA

```
Initialize the parameters popsize, Max_iteraition;
Initialize the positions of slime mould X_i(i = 1,2, ..., n) (4.1);
While (t ≤ Max_iteraition)
    Calculate the fitness of all slime mould;
    update bestFitness, X_b
    Calculate the W by Eq. (4.5);
    For each search portion
        update p, vb, vc;
        update positions by Eq. (4.7);
    End For
    t = t + 1;
End While
Return bestFitness, X_b;
```

Figura 2: Pseudo-algoritmo SMA, Li et al. 2020.

The SMA models itself after the structure of the acellular slime mold Physarum polycephalum, actively behaving in search of a food source. Hereafter, Physarum polycephalum will be referred to as "Slime Mold." This complex entity exhibits an amoeboid structure composed of interconnected tubes responsible for transporting cytoplasm throughout the organism. This capability is made possible by the mold's peculiar anatomy, allowing the construction of complex networks of veins between various food sources, enabling simultaneous feeding from all of them. Upon locating a food source, the Slime Mold's biochemical oscillator sends contractile waves through the venous system, resulting in tubular veins that transport cytoplasm. The speed of cytoplasmic flow is related to the thickness of the vein wall, increasing as the flow rate increases and decreasing as the flow rate decreases. The slime mold relies on both positive and negative feedback to reach food sources. In the following, detailed explanations about the proposed mathematical model and the associated procedure will be provided.

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

The formula below allows the searching individual to explore all possible directions near the optimal solution, thus simulating the structure of the circular sector of mold as it approaches food.

$$\vec{X}(t+1) = \begin{cases} \vec{X_b}(t) + \vec{vb}(\vec{W}.\vec{X_A}(t) - \vec{X_B}(t)), r < p \\ \vec{vb}.\vec{X}(t), r \geq p \end{cases} \quad \text{Eq. (4.1)}$$

t: represents the iteration;

$\vec{X_b}$ **:** represents the location with the highest odor concentration currently found

$\vec{X}$ **:** Represents the location of the Slime Mold

$\vec{X_A}$ *e* $\vec{X_B}$: Represents two randomly selected individuals from Slime Mold

p = tanh $|S(i) - DF|$, where i $\in$ (1, 2, …, n) Eq. (4.2)

$\vec{vb}$ **:** [−a, a] Eq. (4.3)

a: arco (- $\left(\frac{t}{max_t}\right)$ + 1) Eq. (4.4)

S(i) Represents fitness (aptitude) of $\vec{X}$

DF: Representation of the best fitness resulting from all interactions

$\vec{W}$ **:** represents the weight of the Slime Mold

The formula below mathematically simulates the positive and negative feedback between the vein width of the slime mold and the concentration of food that has been explored, where:

$$\vec{W}(\text{SmellIndex}) = \begin{cases} 1 + r.log(\frac{bF - S(i)}{bF - wF} + 1).\ current\ iteration \\ 1 - r.log(\frac{bF - S(i)}{bF - wF} + 1).\ other \end{cases} \quad \text{Eq. (4.5)}$$

SmellIndex = sort(S) Eq. (4.6)

*log***:** Used to smooth the rate of change of the numerical value so that the contraction frequency value does not change drastically.

Current Condition: Simulates the slime mold to adjust its search patterns based on the quality of food sources.

Others: When a high concentration of food is encountered, the weight near that region increases. Conversely, when the food concentration is low, the weight of the region is reduced, encouraging exploration of other areas.

r **:** Denotes a random value within the range of [0, 1], representing the uncertainty in the venous contraction mode.

*bF***:** Denotes the best fitness achieved in the current iterative process.

*wF***:** Denotes the worst fitness value obtained in the current iterative process.

$\vec{vc}$: Decreases linearly from 1 to 0.

When a high concentration of food is encountered, a greater weight is placed on that region, making it more likely for the slime mold to move towards it. Conversely, when the food concentration is low, the weight of the region is reduced, prompting the slime mold to explore other areas.

Based on this principle, the mathematical formula for updating the location of the slime mold is as follows:

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

$$X^{\rightarrow} * = \begin{cases} rand\ .(UB - LB) + LB, rand < z \\ X_b^{\rightarrow}(t) + vb^{\rightarrow}.(W.X_A^{\rightarrow}(t) - X_B^{\rightarrow}(t)), r<p \\ vc^{\rightarrow}.X^{\rightarrow}(t),\ r \geq p \end{cases} \quad \text{Eq. (4.7)}$$

Where LB and UB represent the lower and upper bounds of the search space, rand and r indicate random values within the range [0,1].

The z value represents a probability balance between exploration and exploitation. As the algorithm aims to simulate the exploration of other areas in search of a higher quality food source, rather than investing everything in a single source, the value of $v\vec{b}$ oscillates randomly between [-a, a] and gradually approaches zero as iterations increase. The value of $v\vec{c}$ oscillates between [-1,1] and eventually tends towards zero. The synergistic interaction between $v\vec{b}$ and $v\vec{c}$ mimics the selective behavior of slime mold.

Slime mold primarily relies on the wave propagation process generated by the biological oscillator to modulate cytoplasmic flow within its veins, seeking to position itself advantageously in relation to food concentration. The mold can approach food sources more efficiently when identifying higher quality, adopting a more gradual approach when food concentration is lower in a given location. This strategy enhances the slime mold's efficiency in selecting the optimal food source. Even after finding a higher quality food source, the mold still reserves a portion of its organic resources to explore other areas, rather than concentrating all its efforts on a single source [Li et al. 2020].

5. Data of the Studied Problem

This section presents a synthesis of the data regarding student pickup points, assessed vehicles, and other parameters applied to the school transportation problem in the proposed fictional example of school bus routing in the city of Caraguatatuba.

Specifically, the study aims to optimize the school transportation routes for the Association of Parents and Friends of Exceptional Children (APAE) in Caraguatatuba. This particular case considers that transportation is intended to pick up each student from their residence. Therefore, the studied problem can be modeled as the Multiple Traveling Salesman Problem (mTSP), a generalization of the Traveling Salesman Problem that allows for more than one salesman, i.e., multiple vehicles leaving from the same school to pick up students.

The data is based on information provided by the city hall website, however, certain values are abstracted to formulate the problem.

The set of elements and parameters used in the model are related as listed below:

Sets:

- Schools: Set of schools to be served by school buses;
- Students: Set of students who need to be transported to school;
- Buses: Set of school buses available.

Parameters:

- Maximum Vehicle Capacity: The capacity of the selected vehicle type.
- Distance: A matrix representing the distances between schools and bus stops.
- Initial Location: The starting point of the buses, presumably the school.

The problem aims to minimize the total distance traveled by the fleet of vehicles selected for transportation. To achieve this, four different vehicle types will be evaluated:

- Cars: Capacity of 4 students;
- Minivans: Capacity of 6 students;
- Vans: Capacity of 12 students;
- Minibusses: Capacity of 22 students.

A total of 20 students located in the vicinity of the institution will be considered. Their locations are fictional and randomly generated, as shown in Figure 3.

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

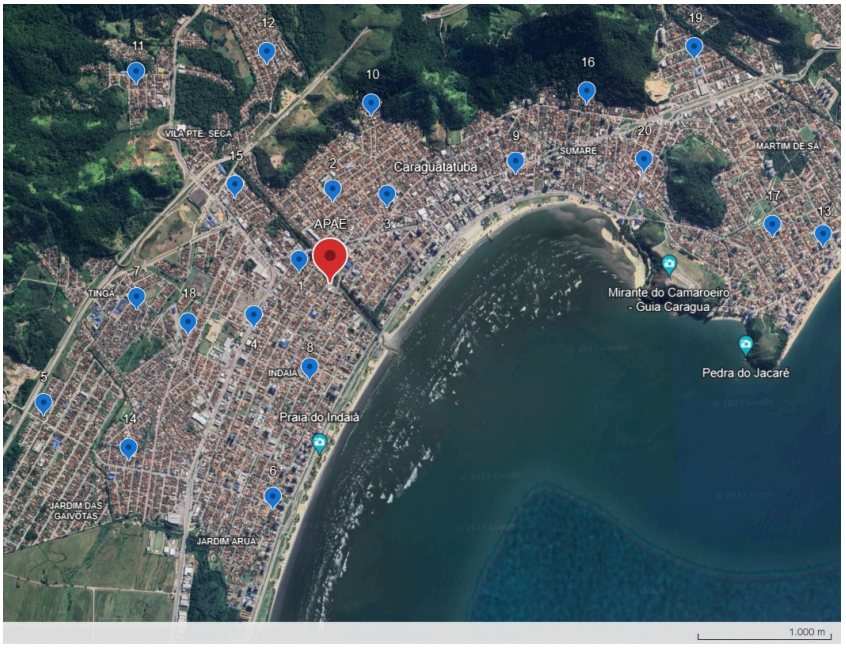


Figure 3: Map of the fictional location of 20 students.

Using the latitude and longitude coordinates of each fictional student's home, a vehicle route was generated using Google Earth, as shown in Figure 4. This provided a realistic approximation of the distance (in meters) between each vehicle stop, with the school designated as point 0.

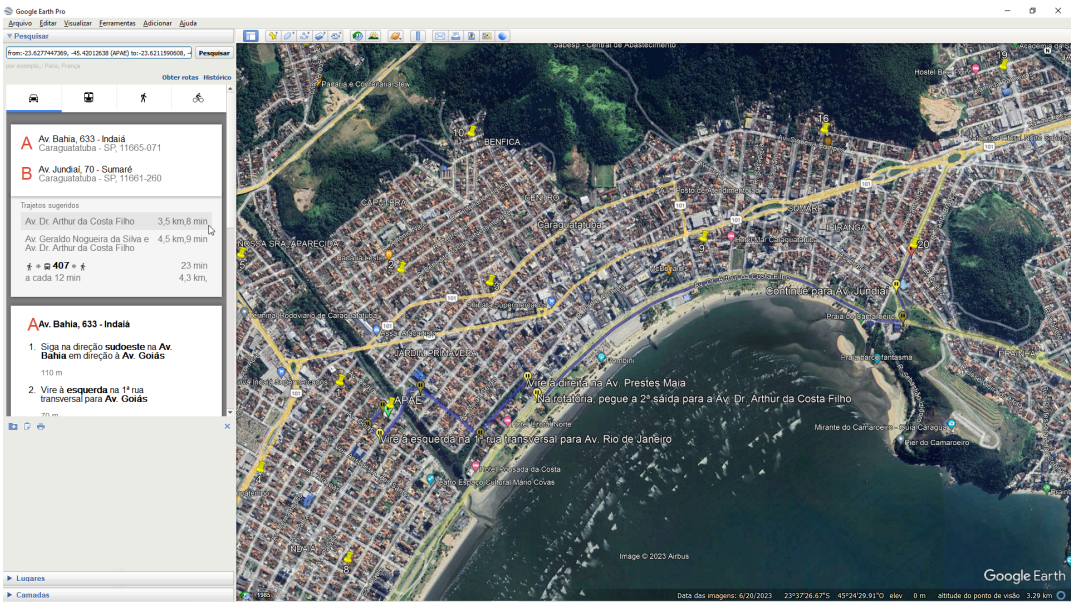


Figure 4: Illustration of how distance values between points were obtained using Google Earth.

To evaluate the performance of the proposed model, variations in parameters will be made, generating problems 1 through 13, as detailed in Table 1. The processing time will be assessed as the number of students increases, along with determining which vehicle type yields the shortest total distance traveled for each student quantity considered.

6. Mathematical Model

To solve the exact problem, the Multiple Traveling Salesman Problem (mTSP) model applied to the Vehicle Routing Problem (VRP) was utilized.

Following the VRP formulation [Arenales et al. 2015], consider a complete directed graph G = (N, E).

Sets:

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

C = {1, ..., n} = Set of nodes, representing students to be served by school transportation.

N = C ∪ {0, n+1} = {0, n+1} represent the school.

E = {(i, j) : i, j ∈ N, i ≠ j, i ≠ n+1, j ≠ 0} = Set of edges between the served points. Routes start at 0 (school) and end at n+1 (school).

K = Set of available school vehicles, where k ∈ K.

Parameters:

c_{ij} = Distance between points i and j, (i, j) ∈ E.

Q_m = Maximum capacity of vehicle k ∈ K.

Decision Variables:

x_{ijk} = Binary variable, k ∈ K, (i, j) ∈ E. It equals 1 if bus k travels from stop i to j, and 0 otherwise.

Objective Function:

Minimize the total cost, which is the sum of the distances traveled by all buses:

$$Minimize\ \Sigma_{k\in K}\Sigma_{(i,j)\in E}\, c_{ij}\, x_{ijk}$$

Constraints:

Ensure each student i is served by a single vehicle:

$$\Sigma_{k\in K}\Sigma_{j\in N\setminus\{0\}}\, x_{ijk} = 1,\ i \in C$$

Ensure vehicle k does not exceed its capacity Q:

$$\Sigma_{j\in N\setminus\{0\}}\, x_{ijk} \leq Q,\ k \in K$$

Ensure each vehicle k departs from the school only once:

$$\Sigma_{j\in N\setminus\{0\}}\, x_{0jk} = 1,\ k \in K$$

Ensure each vehicle k returns to the school only once:

$$\Sigma_{j\in N\setminus\{0\}}\, x_{i,n+1,k} = 1,\ k \in K$$

Ensure each vehicle k enters and leaves a node h exactly once:

$$\Sigma_{i\in N\setminus\{n+1\}}\, x_{ihk} - \Sigma_{j\in N\setminus\{0\}}\, x_{hjk} = 0, h \in C,\ k \in K$$

Eliminate sub-tours:

$$\Sigma_{i\in S}\Sigma_{j\in S} x_{ijk} \leq |S| - 1\,, S \subset C,\ 2 \leq |S| \leq |n/2|\,,\ k \in K$$

Variable Type:

$$x \in K^{\mathrm{K|E|}}$$

7. Heuristics for the Vehicle Routing Problem in School Transportation:

7.1 Genetic Algorithm

The algorithm randomly generates a population of solutions, with a parameter of 1,000 solutions used in this case. Schools already present in the routes of the "offspring" are excluded. A crossover probability rate of 0.9 is set. A mutation function is defined that introduces a small probability of alteration in vehicle routes. Mutation can involve swapping schools within a route, reordering schools, or making adjustments to existing routes. A mutation probability of 0.05 was established. This solution represented the most efficient routing of vehicles to serve schools in the context of school transportation.

7.2 Slime Mold Algorithm Metaheuristic

For the SMA, a Python implementation was utilized, employing the "mealpy" library. The SMA class was instantiated with specific parameters: 1000 iterations (epochs), an initial population of 50 agents (pop_size), and a probability threshold of 0.03 (p_t). Model training occurs through the solve() method, which uses parallelism during the generation of the initial population and the update of targets for the entire population. Throughout the iterations, the SMA seeks to optimize the objective function by efficiently exploring the search space defined by established boundaries. The training results are printed, displaying the solution and fitness value of the best agent found during the optimization process.

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

The data files and models are available on the source code hosting and version control platform, Github, at the following link: [https://github.com/Bruunoreis/sma_ga_mtsa.git].

8. Results

8.1 Exact Model

The exact model was solved using the Gurobi solver, with the solution obtained through the NEOS Server. The solution was limited to 20 points due to the limited capacity available through the tool.

As the number of students increases, the number of variables and constraints grows such that the problem becomes computationally expensive to solve. Therefore, it becomes infeasible to solve a real-world problem where the number of students reaches the order of hundreds per school. Consequently, applying other methods to solve the problem is necessary, obtaining a solution close to the optimal solution, given the limitations in time and computational resources for achieving a satisfactory solution.

Since the exact solution provided by the NEOS server does not support more than 20 students, beyond this number, comparison with the metaheuristic solution is not possible. Hence, the comparison evaluated the response for up to 20 students. From this point onwards, metaheuristics prove essential for obtaining a solution to the problem.

Figure 5 below illustrates an example of the graphical representation of the optimal route for the case of 5 students and one vehicle to demonstrate the solution of the problem using the exact model. The vehicle departs from the school (0) towards point 2-3-5-4-1 and returns to the origin school.

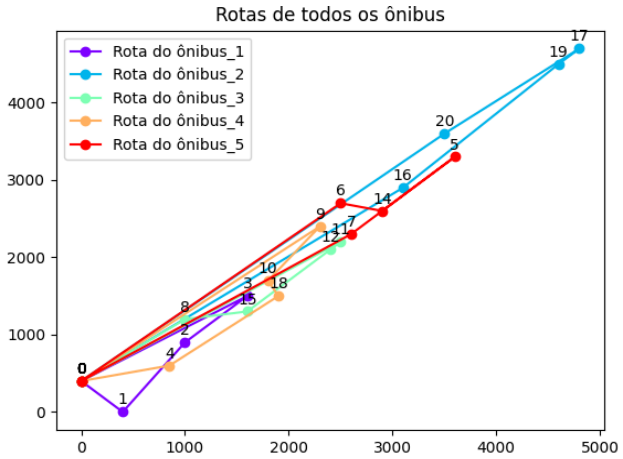


Figure 5: Solution by the exact model for the instance of Problem 1: 0-2-3-5-4-1-0

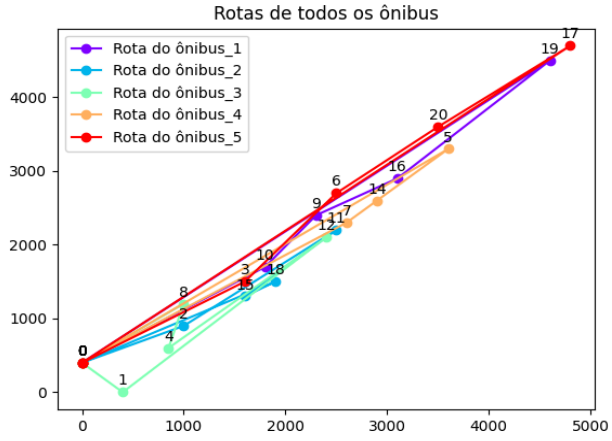
8.2 Metaheuristics

The metaheuristics were validated by comparing them to the exact solution provided by the mathematical model. In the comparison between the metaheuristics, it was observed that the GA presented better results than the SMA in all instances. This discrepancy can be explained by the formulation of the metaheuristics.While SMA, inspired by biological processes, tends to enhance the search for the best route based on the concentration of slime, strengthening connections along paths with higher concentrations and seeking an efficient path [Adamatzky et al., 2013], the genetic algorithm follows an evolutionary approach. It uses concepts of natural selection, crossover, and mutation to evolve a population of solutions over generations, eliminating less efficient solutions. Therefore, slime mold focuses on amplifying promising routes based on local conditions, namely slime concentration, while the genetic algorithm seeks global optimization, introducing variation and selection over iterations to converge towards more efficient solutions [Chen et al., 2010].

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br



A) Solução GA: 41050 (20 alunos, 5 ônibus)



B) SMA Solution: 5070 (20 students, 5 buses)

Although the GA consistently outperformed the SMA in the provided results, it's important to note that the SMA still managed to generate feasible routes for the given problem. The SMA's successful application in finding the shortest metro paths in Tokyo [Tero et al., 2010] and Beijing [Adamatzky et al., 2013] highlights its potential in solving routing problems. The lack of existing research on vehicle, bus, or school bus routing using the SMA makes this work potentially pioneering in exploring this specific application.

8.3 Comparison of Results:

Table 1 presents the comparative data between the solutions found by the Gurobi solver, SMA, and GA.

Table 1: Comparison between the results of the exact model and the GA and SMA Metaheuristics

Instance				Gurobi		Genetic Algorithm Metaheuristic - GA			Metaheuristic Smile Mold Algorithm -SMA		
inst.	**Students**	**Vehicle Capacity**	**Number of vehicles**	**Optimal solution (m)**	**Execution time(s)**	**Best Solution**	**Average**	**Execution Time(s)**	**Best Solution**	**Average**	**Execution Time(s)**
1	5	4	2	3800	2	11070	11070	14s	11450	11450	7s
2	10	4	3	12500	9	13632	13640	15s	13200	15150	15s
3	15	4	4	30100	1020*	21941	21959	17s	38059	40150	27s
4	20	4	5	**	**	39000	40550	41s	50700	54250	1225s
5	5	6	1	3800	2	6539	6539	8.7s	10600	10600	4s
6	10	6	2	11650	21	15200	15200	23s	14900	16150	10s
Instance				**Gurobi**		**Genetic Algorithm Metaheuristic - GA**			**Metaheuristic Smile Mold Algorithm -SMA**		

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

inst.	Students	Vehicle Capacity	Number of vehicles	Optimal solution (m)	Execution time(s)	Best Solution	Average	Execution Time(s)	Best Solution	Average	Execution Time(s)
7	15	6	3	29000	300	28550	28550	16s	36000	36000	20s
8	20	6	4	**	**	21666	21789	17s	37200	38150	13s
9	5	12	1	3800	2	6539	6539	8.7s	10600	10600	4s
10	10	12	1	11650	21	15350	16850	3.9s	17300	15750	7s
11	15	12	2	27500	120	32550	32550	16s	37200	38150	13s
12	20	12	2	**	**	42250	42250	17s	49200	49200	440s
13	20	22	1	26600	420	42800	42800	40.5s	52700	52700	12s

Observations:

** The execution time increased abruptly due to the growing number of variables in the problem. In this case, the addition of one more vehicle to the route significantly impacted the solving time.*

***NEOS lacks the capacity to handle the increased number of variables, resulting in the termination message: "Your job was terminated because it exceeded the maximum allocated memory for a job."*

The limitation of 20 study points for the exact solution through NEOS highlights the crucial need for a metaheuristic approach to solve real-world problems with larger instances.

9. Conclusion

The exact solution in AMPL provided by the NEOS server did not support more than 20 students. From this point on, metaheuristics proved essential to obtain a solution to the problem.

In the comparison between the Metaheuristics, it was analyzed that the GA presented better results than the SMA in all instances. This divergence can be explained in the formulation of the metaheuristics. While the SMA, inspired by biological processes, tends to enhance the search for the best route based on the concentration of slime, strengthening connections along the path with higher concentration, looking for an efficient path, the genetic algorithm follows an evolutionary approach, using concepts of natural selection, crossover, and mutation to evolve a population of solutions over generations, eliminating less efficient ones.

However, the Slime Mould Algorithm was able to solve the problem, presenting viable routes for the work in question. As SMA is relatively recent in the literature, so far its application is being studied in several problem domains. This study is part of these efforts to evaluate the algorithm for new applications, in this case related to the routing of vehicles, buses or school buses using SMA. Therefore, this work aimed to evaluate the application of this technique related to transportation problems.

References:

Arenales M., Armentano V., Morabito R., Yanasse H.. (2015). Pesquisa Operacional: Para Cursos de Engenharia, Segunda Edição. Editora Elsevier. Rio de Janeiro.

BRASIL. [Constituição (1988)]. Constituição da República Federativa do Brasil de 1988. Brasília, DF: Presidente da República, [2016].

Burchett, Joseph N et al (2020). Revealing the Dark Threads of the Cosmic Web.The Astrophysical Journal Letters, 891:L35 (12pp), 2020 March 10. https://doi.org/10.3847/2041-8213/ab700c

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br

Caldas L., Martinelli R., Rosa B. (2022), Solving a School Bus Routing Problem in Rural Areas: An Application in Brazil. In: de Armas, J., Ramalhinho, H., Voß, S. (eds) Computational Logistics. ICCL 2022. Lecture Notes in Computer Science, vol 13557. Springer, Cham. https://doi.org/10.1007/978-3-031-16579-5_12

Chen, S. e outros. al.(2010). Otimização baseada em algoritmo genético para redes de rádio cognitivas", Simpósio IEEE Sarnoff.

Dantzig, G.; Fulkerson, R.; Johnson, S. (1954), Solução de um problema de caixeiro viajante em grande escala. Jornal da sociedade de pesquisa operacional da América, INFORMS, v. 4, pág. 393–410, 1954.

Gutin, G. e Punnen, A. P., (2002). The traveling salesman problem and its variations. Combinatorial optimization. Kluwer Academic, Dordrecht, London. Dantzig, G. B. and Ramser, J. H. (1959). The Truck Dispatching Problem. ManagementScience, 6(1):80–91. Lenstra, J. K. and Rinnooy Kan, A. H. G. (1981). Complexity of Vehicle Routing andScheduling Problems. Networks, 11:221–227.

Giaglis, G. M. et al. (2004). Minimizing logistics risk through real-time vehicle routingand mobile technologies: Research to date and future trends. International Journal ofPhysical Distribution & Logistics Management, 34(9):749–764.

Gharehchopogh FS, Ucan A, Ibrikci T, Arasteh B, Isik G. (2023), Slime Mould Algorithm: A Comprehensive Survey of Its Variants and Applications. Arch Comput Methods Eng. 2023;30(4):2683-2723. doi: 10.1007/s11831-023-09883-3. Epub 2023 Jan 12. PMID: 36685136; PMCID: PMC9838547.

Li L., Fu Z. (2002), The school bus routing problem: a case study. The Hong Kong Polytechnic U. Article in Journal of the Operational Research Society.

Li S., Chen H., Wang M., Heidari A. A. (2020), Slime mould algorithm: A new method for stochastic optimization. Future Generation Computer Systems. https://aliasgharheidari.com/SMA.html. 300-323. 10.1016/j.future.2020.03.055.

Manumbum D. M., Mujuni E., Kuznetsov D. (2014), Mathematical Formulation Model for a School Bus Routing Problem with Small Instance Data. ISSN 2224-5804 (Paper). Mathematical Theory and Modeling E-Journal. Vol.4, No.8.

Moraes, M. B. da C. e Nagano, M. S. (2013), Cash management policies by evolutionary models: a comparison using the Miller-Orr model. JISTEM - Journal of Information Systems and Technology Management [online]. 2013, v. 10, n. 3 [Accessed 10 June 2023], pp. 561-576. Available from: <https://doi.org/10.4301/S1807-17752013000300006>. ISSN 1807-1775.

Melo I. E. S., Kramer R. (2021), Uma abordagem heurística para o Multi-Period School Bus Routing Problem. Simpósio Brasileiro de Pesquisa Operacional.

Porto M. F., De Carvalho I. R. V., Teixeira R., Cardoso L., Nunes N. T. R., Alves L. V. R., Baracho R. M. A. (2018), Benefícios Do Transcolar Rural Na Otimização De Rotas: Um Estudo Comparativo Entre Diferentes Ferramentas De Roteamento. 8º CONGRESSO LUSO-BRASILEIRO PARA O PLANEAMENTO URBANO, REGIONAL, INTEGRADO E SUSTENTÁVEL (PLURIS 2018). Coimbra - Portugal.

Tero, A. & Takagi, S. & Saigusa, T. & Ito, K. & Bebber, D. & Fricker, M. & Yumiki, K. & Kobayashi, R. & Nakagaki, T.. (2010). Rules for Biologically Inspired Adaptive Network Design. Science (New York, N.Y.). 327. 439-42. 10.1126/science

DOI: 10.59254/sbpo-2024-193382
https://proceedings.science/p/193382?lang=pt-br