An Evolutionary Approach for the Optimal Design of Public Transportation Networks

Julia Ramos Beltrão
Instituto de Computação - UNICAMP
Av. Albert Einstein, 1251 - Cidade Universitária, Campinas/SP
julia.beltrao@students.ic.unicamp.br

Fábio Luiz Usberti
Instituto de Computação - UNICAMP
Av. Albert Einstein, 1251 - Cidade Universitária, Campinas/SP
fusberti@ic.unicamp.br

ABSTRACT

An Integer Linear Programming (ILP) model for the Single Line Planning Problem - a simplified version of the Line Planning Problem - has been previously proposed in the literature. Although this approach is able to easily find optimal solutions for small instances, it fails to scale. In this work, we propose a method based on the Multi-Parent Biased Random Key Genetic Algorithm with Implicit Path-Relinking (BRKGA-MP-IPR) metaheuristic and compare to the results obtained by the ILP formulation. Our experiments revealed the BRKGA-MP-IPR is able to outperform the ILP solution for large instances, indicating the two approaches to be complementary.

KEYWORDS. Line planning problem. Network design. Genetic algorithms.

Logistics and transport, metaheuristics, mathematical programming

https://proceedings.science/p/157583?lang=pt-br

1. Introduction

The importance of public transportation in the history of humankind is undeniable. Its development has favored cities to attain the level of urbanization they have today; it aids people to commute to work on a daily basis; it has "shortened" distances in a way that could not be thought of 200 years ago. Its relevance nowadays is reassured by the fact that the number of trips carried out on public transportation had an 18% increase between the years 2000 and 2015 [UITP, 2017]. Moreover, public transport is the answer to several problems large cities face in the present day, such as traffic congestion, air pollution, and energy consumption [Kepaptsoglou and Karlaftis, 2009]. Therefore, it is important that public transit is adequately financed and planned in order to guarantee affordable and efficient mobility to citizens.

Many approaches to automatize the process of public transit design have been proposed in the literature. [Laporte et al., 2007] proposed a model based on linear programming to design networks of rapid-transit modes - which are commonly segregated and independent of the street network. The model is rather complex, taking into account travel demand, how this demand is distributed between private and public transport, the decision of constructing both stations and alignments connecting stations - as well as the costs of construction.

[Borndörfer et al., 2007] proposed a column-generation approach to network design tied to the problem of finding the frequencies of each line (this combination is known as the line planning problem) in order to satisfy a given travel demand. They detected two objectives: minimizing operating costs for the company and minimizing traveling times for passengers.

[Bussieck et al., 2004] also investigated the line planning problem. They used both nonlinear and mixed integer optimization to produce models for the problem and considered two objectives: minimizing operating costs and passenger convenience, which is modeled as a minimum for the frequency.

[Ángel Marín and Jaramillo, 2009] presented a model for urban rapid transit network design, which comprises of the location of stations and of the alignments between stations. The model aims at maximizing travel demand supplied under cost constraints and it also takes into account that users can choose their transportation modes and trips. As the model cannot be solved efficiently by a Branch and Bound due to the large number of variables and constraints, algorithms based on Benders decomposition were proposed.

[Zarrinmehr et al., 2016] proposed a multi-objective bi-level optimization model for the network design problem. The model takes into consideration (maximizing) transit ridership, (minimizing) agency operating cost, system performance and an elastic demand. The authors conceived a greedy algorithm to solve the model.

[Beltrão and Usberti, 2020] proposed an Integer Linear Programming (ILP) formulation to model the drafting of the itinerary of a single public transit network line, which they named the Single Line Planning Problem (SLP) - a particularization of a problem that is usually referred in the literature as the line planning problem. The ILP formulation maximizes the demand served, that is, the number of trips carried out on the line.

In this work, we propose a solution using the BRKGA-MP-IPR metaheuristic to search for good feasible solutions to the Single Line Planning Problem and compare those results to the ones obtained by the ILP model, as presented in [Beltrão and Usberti, 2020], for the same instances used in their work. In Section 3 we present the results obtained in the previous work and introduce those obtained by our experiment with the BRKGA-MP-IPR metaheuristic, as well as discuss how the methodologies seem to present complementary contributions depending on the type of instance.

https://proceedings.science/p/157583?lang=pt-br

A fairly new method, BRKGA-MP-IPR has few applications reported in the literature. [Kummer et al., 2022] used BRKGA-MP-IPR address the home health routing and scheduling problem. [Londe et al., 2021] studied the RSI allocation problem using exact and heuristic methods, the BRKGA-MP-IPR being one of them. [Andrade et al., 2022] adapted the BRKGA-MP-IPR framework to solve the physical cell identity assignment problem. In all cases, BRKGA-MP-IPR is used in order to improve the scalability of optimization problems.

Our contribution The exact method proposed in [Beltrão and Usberti, 2020] struggles to solve instances of 64 nodes and over when subjected to a tight budget constraint. We were successful in developing a decoder for the BRKGA-MP-IPR metaheuristic in order to address this issue.

2. Proposed Solution

In this section, we explain the functioning of the metaheuristic of our choice and the specificities of our solution to the problem.

2.1. The BRKGA-MP-IPR metaheuristic

BRKGA

Proposed in [Gonçalves and Resende, 2011], *Biased Random-Key Genetic Algorithm* or *BRKGA* is an evolutionary metaheuristic. In this metaheuristic, a population of individuals is evolved according to the principle of the survival of the fittest. Each individual is represented by a chromosome Q with n genes; each gene is a rational number uniformly drawn from the interval [0,1]. A decoder algorithm, which is specific to the problem being studied, is responsible to map a chromosome Q to a solution S of the problem. BRKGA starts with an initial population of p randomly generated individuals (chromosomes).

The metaheuristic then proceeds to repeat some steps for g iterations, which are named generations. In a given generation ζ, the chromosomes are mapped into solutions, those being used to separate the population into two different groups: elite and non-elite. The elite group holds the p_e fittest individuals, that is, the chromosomes that map to the p_e best solutions - which are passed on to generation $\zeta + 1$; the non-elite group is comprised by the remaining individuals of the population, including p_m randomly generated individuals, named mutants. The other $p - (p_e + p_m)$ individuals are generated through crossover, which produces a new individual by sampling genes from two individuals of generation $\zeta - 1$, called parents; those are drawn randomly from the elite and non-elite groups, one from each. A gene r for the new chromosome is chosen from the elite parent with probability ρ_e. After g generations, BRKGA returns the best chromosome Q^* and the corresponding solution S^*.

BRKGA With Multiple Parents

Proposed by [Lucena et al., 2014], *Multi-Parent Biased Random-Key Genetic Algorithm* or *BRKGA-MP* is a variation of the BRKGA metaheuristic in which more than two parents are used to generate a new offspring.

Path-Relinking

The idea behind the Path-Relinking approach is to explore the neighborhood of the path between two feasible solutions. Two feasible solutions - a base and a guide solution - are sampled from the pool of known feasible solutions. These solutions need to have a minimum value of different components between them; such value is fixed beforehand. Then a component of the base solution

https://proceedings.science/p/157583?lang=pt-br

is selected and replaced with a component from the guide solution. This process is repeated until all components of the base solution have been substituted by components from the guide solution. All the intermediate solutions are evaluated and the best one is returned [Glover et al., 2000].
The drawback of this approach is that it is strongly dependent of the specific problem at hand. The use of Path-Relinkig combined with BRKGA was first proposed in [Ribeiro et al., 2021].

Implicit Path-Relinking
[Andrade et al., 2021] proposed a variation to the Path-Relinking strategy, which they named *Implicit Path-Relinking*. This variation takes advantage of the already implemented chromosome decoder. Two different versions of Implicit Path-Relinking were proposed, *Direct Implicit Path-Relinking* and *Permutation-Based Implicit Path-Relinking*. For this work, we are interestered in the latter.

Permutation-Based Implicit Path-Relinking
Crafted to address problems where the order of the genes in the chromosome are used by the decoder to create a solution, in this this version of Implicit Path-Relinking, instead of exchanging keys between the base and guide solutions, we use the permutation from the guide solution to switch the keys of the base solution in such a way that they induce the same "sub-permutation" on the base solution [Andrade et al., 2021].

2.2. Our Decoder
We consider the chromosome to be ordered. Each gene represents a node of the network, according to its position in the chromosome. The first gene represents node 1, the second gene represents node 2, and so forth, up to gene n representing node n, where n is both the number of genes in the chromosome and of nodes in the network. Each gene has an associated value named key; we use those to build a path that is a valid solution to the SLP problem.
We start by adding to the path the node represented by the gene with the lowest key. Let's call this node v. We continue building the path by adding to it the node among its neighbors with the lowest key. Let's call it u. We keep growing the path by adding the node among the union of neighbors of v and u that has the smallest key value. Let's call this one w. Note that if the path we have so far is (u, v) and w is a neighbor of v, the path resulting of the addition of w is the path (u, v, w); otherwise, the resulting path is (w, u, v). If w is neighbor to both v and u, we add w to the beginning of the path by default - as we did when adding u to the path.
Generally speaking, given a path $p = (v_1, v_2, ..., v_n)$, we search among the neighbors of v_1 and v_n for the node with the smallest key, let us call it z; z is added to p only if the cost of the resulting path does not surpasses the budget, and we do so according to the following rules:
1) if z is neighbor of v_1, we add it at the begging of the path, regardless of z also being a neighbor of v_n or not;
2) if z is only neighbor of v_n, we add z at the end of the path.

3. Experiments
3.1. Computational Environment
All experiments were conducted on a headless machine with two Intel Xeon CPU E5-2630 v4 2.20GHz processors, with 10 cores each, running Ubuntu 18.04.3 - the same computational specifications used in [Beltrão and Usberti, 2020]; the experiments were developed in Python 3.7, using BRKGA-MP-IPR tools implemented in a module named *brkga-mp-ipr* [Andrade et al., 2021].

3.2. Instances

For these experiments we worked with the same instances used in [Beltrão and Usberti, 2020]. Twenty different grids of varying sizes and densities were created. Costs of edges and values of the OD matrix were generated using uniform distributions.

3.3. Parameters

For our experiments we used the following BRKGA-MP-IPR parameters

- *population size:* 2000
- *number of independent populations:* 3
- *mutant percentage in population:* 15%
- *elite percentage in population:* 30%
- *number of elite chromosomes exchanged from each population:* 2
- *interval at which elite chromosomes are exchanged:* 200 seconds
- *number of parents to mate:* 3
- *number of elite parents:* 2
- *bias function to be used to rank the parents:* inverse logarithm
- *minimum distance between chromosomes to path-relink:* 0.15
- *stop criteria:* time limit (10 minutes)

Note that the time limit of our experiments is the same used in [Beltrão and Usberti, 2020].

4. Results and Discussion

In Tables 1, 2 and 3, we reproduce the results obtained by [Beltrão and Usberti, 2020] and present the ones from our experiments with BRKGA-MP-IPR, as described in Section 3.

Taking a closer look at Table 1, one can observe that for smaller networks - the networks with 16 and 36 nodes - the BRKGA-MP-IPR approach underperforms the exact method; that is, while the solver for the exact method is able to find the optimal solution, the best solution the metaheuristic obtains is far apart from the optimal one. For the large networks with low density - 20%, 40% and 60%, and 20% and 40% for networks with 64 and 100 nodes, respectively - BRKGA-MP-IPR also was outperformed by the model. However, for the remaining networks, for which the exact method is not able to find the optimal solution - only lower and upper bounds - during the time limit, the BRKGA-MP-IPR method is able to decrease the gap between those bounds. More precisely, in the case of the 100-node network with 100% density, our heuristic is responsible for defining a lower bound, as the solver was not able to do so within the time limit imposed in the original work.

In Figure 1, we summarize that information by computing the gap between the best feasible solution found by the BRKGA-MP-IPR heuristic and the lower bound found by the exact method - if found - for all networks, under the 25% budget constraint. Figures 2 and 3 do the same for the other two budget constraints. Bars below the *x-axis* indicate that the BRKGA-MP-IPR has done worse than the exact method; bars above it mean the metaheuristic was able to achieve better results than the solver.

https://proceedings.science/p/157583?lang=pt-br

Number of nodes	Results	Density				
		20%	40%	60%	80%	100%
16	LB	605	782	1,926	2,190	3,799
	UB	605	782	1,926	2,190	3,799
	BRKGA-MP-IPR	322	547	1,404	2,190	3,754
36	LB	970	3,767	8,088	11,588	18,581
	UB	970	3,767	8,088	11,588	22,326
	BRKGA-MP-IPR	472	2,845	7,099	11,588	18,357
64	LB	2,391	3,002	23,647	37,720	54,944
	UB	2,391	3,002	23,647	60,509	96,674
	BRKGA-MP-IPR	1,827	2,351	21,351	38,033	56,207
100	LB	1,810	18,009	46,400	86,599	-
	UB	1,810	18,009	95,376	180,947	266,298
	BRKGA-MP-IPR	1,400	16,521	47,784	90,151	135,451

Table 1: Test results for a budget constraint of 25% of the total cost of the network.

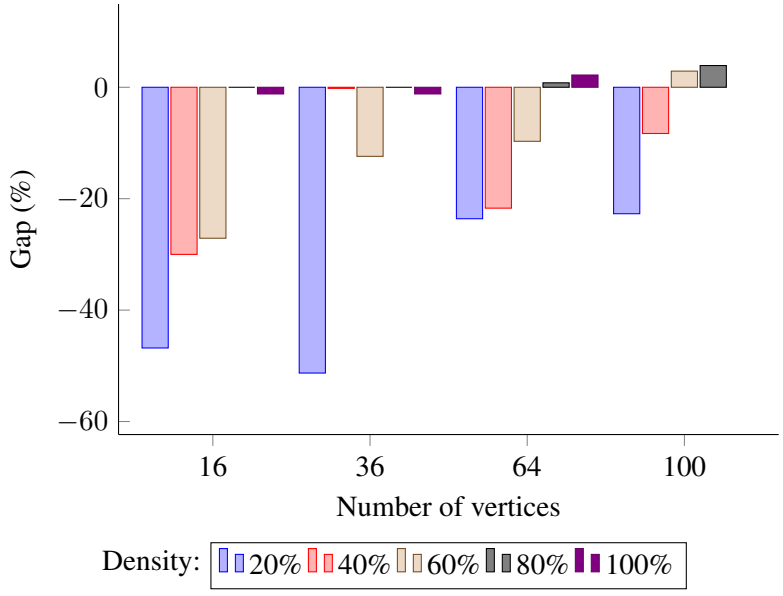


Figure 1: Gap between best feasible solutions of model and BRKGA-MP-IPR, subjected to the 25% budget constraint.

Number of nodes	Results	Density				
		20%	40%	60%	80%	100%
16	LB	984	2,379	4,420	6,911	10,459
	UB	984	2,379	4,420	6,911	10,459
	BRKGA-MP-IPR	716	1,861	3,459	6,911	10,459
36	LB	1,375	5,956	20,839	42,829	58,575
	UB	1,375	5,956	20,839	42,829	58,575
	BRKGA-MP-IPR	804	4,940	18,885	36,536	55,708
64	LB	3,133	3,002	55,835	127,169	200,134
	UB	3,133	3,002	55,835	137,879	205,621
	BRKGA-MP-IPR	2,497	2,351	52,257	121,844	181,094
100	LB	1,810	18,009	116,812	320,725	125,342
	UB	1,810	18,009	242,190	372,483	501,178
	BRKGA-MP-IPR	1,400	16,532	115,055	275,214	414,478

Table 2: Test results for a budget constraint of 50% of the total cost of the network.

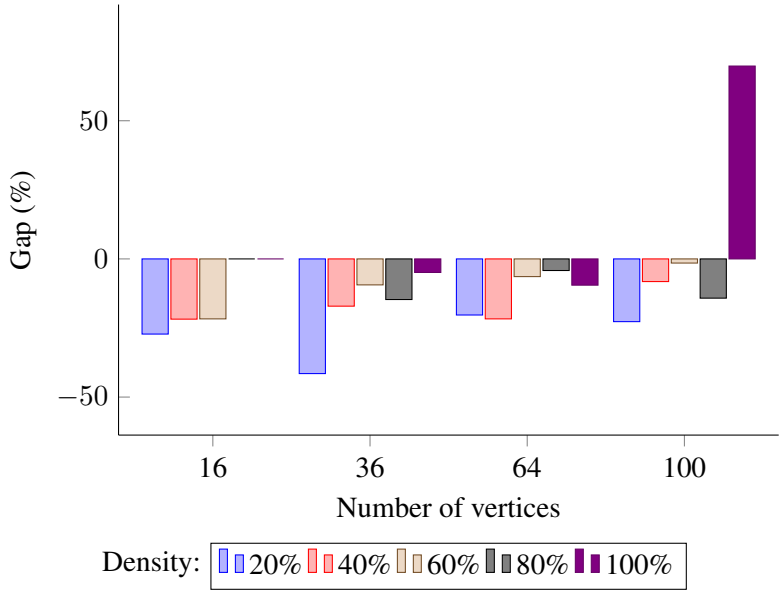


Figure 2: Gap between best feasible solutions of model and BRKGA-MP-IPR, subjected to the 50% budget constraint.

For the 100-node network with 100% density subjected to a budget constraint of 50% of the total network cost, BRKGA-MP-IPR is also able to decrease the gap, very significantly in this case (see Table 2 and Figure 2).

For the 75% budget constraint, the BRKGA-MP-IPR approach was outperformed in every scenario. However, it is likely that, for larger instances than the ones used in this work, the solver would struggle to find solutions.

This behavior can be explained by the fact the model can easily solve the problem for small instances and "loose" budget constraints. The solver starts struggling as we decrease the budget limit and increase the size of the networks - by either nodes or edges. In those cases, the exact method is not able to find an optimal solution within the time limit, only accomplishing to find upper and lower bounds - except for the network with 100 nodes and 100% density subjected to a budget con-

Number of nodes	Results	Density				
		20%	40%	60%	80%	100%
16	LB	984	2,379	4,974	11,926	13,342
	UB	984	2,379	4,974	11,926	13,342
	BRKGA-MP-IPR	716	1,861	3,986	11,796	13,342
36	LB	1,375	5,956	22,659	62,037	65,277
	UB	1,375	5,956	22,659	62,037	65,277
	BRKGA-MP-IPR	804	4,940	20,670	62,037	65,277
64	LB	3,133	3,002	55,835	192,993	206,005
	UB	3,133	3,002	55,835	192,993	206,005
	BRKGA-MP-IPR	2,497	2,351	52,257	175,319	199,784
100	LB	1,810	18,009	125,121	427,150	501,178
	UB	1,810	18,009	125,121	444,717	501,178
	BRKGA-MP-IPR	1,400	16,532	115,055	313,025	416,069

Table 3: Test results for a budget constraint of 75% of the total cost of the network.

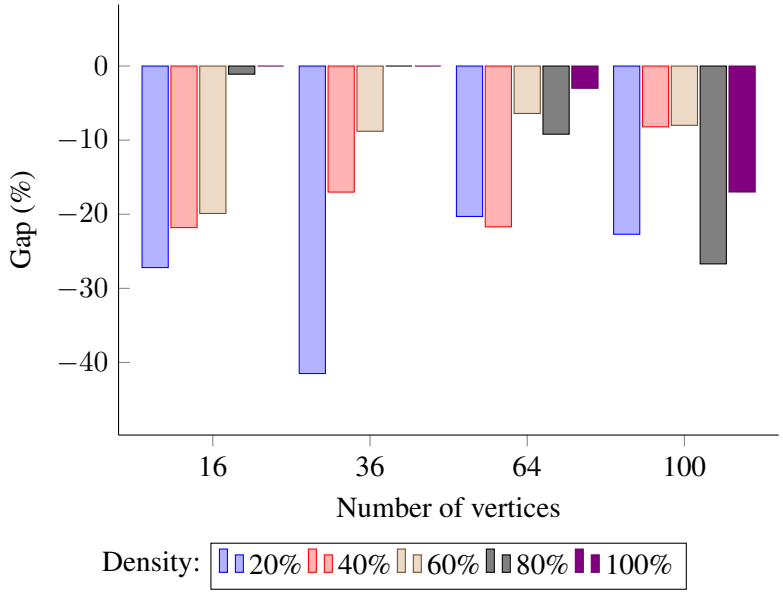


Figure 3: Gap between best feasible solutions of model and BRKGA-MP-IPR, subjected to the 75% budget constraint.

straint of 25% of the cost of the network, for which no lower bound was found.

5. Final Remarks and Future Work

By comparing our results with those obtained by [Beltrão and Usberti, 2020], we could observe that the two approaches seem to be complementary. While the exact method succeeds in finding optimal solutions for small instances or large budget limits, it struggles to scale as the instances increase and the budget limit decrease. On the other hand, the heuristic approach produces non-competitive results for small instances while it manages to provide better solutions than the model for large instances.

In future works, we would like to conduct experiments with larger instances in order to confirm our hypothesis that the two methods in fact complement each other. We would also like to experiment with local search and alternative decoder functions seeking the improvement of the results obtained by the BRKGA-MP-IPR metaheuristic. Additionally, we would like to test providing the solutions obtained heuristically as a warm start for the solver and observe how it impacts its performance.

Acknowledgments

The authors would like to thank the *Brazilian National Council for Scientific and Technological Development* (in Portuguese, *Conselho Nacional de Desenvolvimento Científico e Tecnológico*) for partially funding this work.

References

Andrade, C. E., Pessoa, L. S., and Stawiarsk, S. (2022). The physical cell identity assignment problem: a practical optimization approach. *IEEE Transactions on Evolutionary Computation*.

Andrade, C. E., Toso, R. F., Gonçalves, J. F., and Resende, M. G. (2021). The multi-parent biased random-key genetic algorithm with implicit path-relinking and its real-world applications. *European Journal of Operational Research*, 289(1):17–30. ISSN 0377-2217. URL https://www.sciencedirect.com/science/article/pii/S0377221719309488.

Beltrão, J. R. and Usberti, F. L. (2020). Using optimization to design public urban transportation networks. In *Proceedings of the Brazilian Symposium on Operations Research*.

Borndörfer, R., Grötschel, M., and Pfetsch, M. E. (2007). A column-generation approach to line planning in public transport. *Transportation Science*, 41(1):123–132.

Bussieck, M. R., Lindner, T., and Lübbecke, M. E. (2004). A fast algorithm for near cost optimal line plans. *Mathematical Methods of Operations Research*, 59(2):205–220.

Glover, F., Laguna, M., and Martí, R. (2000). Fundamentals of scatter search and path relinking. *Control and cybernetics*, 29(3):653–684.

Gonçalves, J. F. and Resende, M. G. (2011). Biased Random-Key Genetic Algorithms for Combinatorial Optimization. *Journal of Heuristics*, 17(5):487–525.

Kepaptsoglou, K. and Karlaftis, M. (2009). Transit route network design problem: Review. *Journal of Transportation Engineering*, 135:491–505.

Kummer, A., de Araújo, O., Buriol, L., and Resende, M. (2022). A biased random-key genetic algorithm for the home health care problem. *arXiv preprint arXiv:2206.14347*.

Laporte, G., Marín, Á., Mesa, J. A., and Ortega, F. A. (2007). An integrated methodology for the rapid transit network design problem. In *Algorithmic Methods for Railway Optimization*, p. 187–199. Springer Berlin Heidelberg. ISBN 978-3-540-74247-0.

Londe, M. A., Andrade, C. E., and Pessoa, L. S. (2021). An evolutionary approach for the p-next center problem. *Expert Systems with Applications*, 175:114728.

Lucena, M. L., Andrade, C. E., Resende, M. G., and Miyazawa, F. K. (2014). Some extensions of biased random-key genetic algorithms. In *Proceedings of the 46th Brazilian Symposium of Operational Research*, p. 1–12.

Ribeiro, C. C., Riveaux, J. A., and Brandao, J. S. (2021). Biased random-key genetic algorithms using path-relinking as a progressive crossover strategy. In *2021 5th International Conference on Intelligent Systems, Metaheuristics amp; Swarm Intelligence*, ISMSI 2021, p. 28–36, New York, NY, USA. Association for Computing Machinery. ISBN 9781450389679. URL https://doi.org/10.1145/3461598.3461603.

UITP (2017). Urban public transport in the 21st century. *uitp.org/*. Available: http://www.uitp.org/urban-public-transport-21st-century [Last access: 29 Jan 2020].

Zarrinmehr, A., Saffarzadeh, M., Seyedabrishami, S., and Nie, Y. M. (2016). A path-based greedy algorithm for multi-objective transit routes design with elastic demand. *Public Transport*, 8(2): 261–293.

Ángel Marín and Jaramillo, P. (2009). Urban rapid transit network design: accelerated Benders decomposition. *Annals of Operations Research*, 169(1):35–53.

https://proceedings.science/p/157583?lang=pt-br