The RSI Allocation Problem: exact and heuristic methods

Mariana Alves Londe
Programa de Pós-Graduação em Engenharia de Produção
Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio
R. Marquês de São Vicente, 225 - Gávea, Rio de Janeiro - RJ
mlonde@aluno.puc-rio.br
Defense date: April 2021
Advisor: Luciana de Souza Pessoa
Co-Advisor: Carlos Eduardo de Andrade

RESUMO

Em redes de comunicação sem fio, o Root Sequence Index (RSI) é utilizado para alocar canais entre o equipamento do usuário e a estação rádio-base. A alocação de RSIs com valores próximos a rádios vizinhos pode causar colisões, levando a falhas de serviço e degradação de performance. Neste estudo, a alocação do RSI é modelada como uma generalização do clássico Problema de Coloração de Grafos, indicando que deve existir uma distância mínima entre as cores de dois vizinhos. Para a alocação do RSI, uma distância máxima também é necessária. Este estudo apresenta métodos para alocar o RSI, ao mesmo tempo que minimiza o risco de colisões, para dois modos diferentes de operação encontrados em redes. Os modelos exatos e as metaheurísticas são explorados e comparados em instâncias obtidas de cenários reais.

PALAVRAS CHAVE. Root Sequence Index, Modelos Matemáticos, Algoritmo Genético com Chaves Aleatórias Viciadas

Tópicos: TEL&SI(PO em Telecomunicações e Sistemas de Informações), MH(Metaheurísticas)

ABSTRACT

In mobile wireless networks, the Root Sequence Index (RSI) is used to allocate uplink channels between the user equipment and the base station. The assignment of RSIs close-in-range to neighbor radios may cause collisions leading to failures on service establishment and performance degradation. In this study, we model the RSI allocation as a generalization of the classical Graph Coloring Problem, indicating that there must be a minimum distance for the assigned colors of two neighbors. For the RSI allocation, a maximal distance is also needed. We develop methods for allocating the RSI, while minimizing the risk of collisions, for two different operation modes found on carrier-grade networks. Both exact and heuristic methods are explored and compared. We test the proposed approaches on instances obtained from real-life context.

KEYWORDS. Root Sequence Index, Mathematical Models, Biased Random-Key Genetic Algorithm

Paper topics: TEL&SI (PO in Telecommunications and Information Systems), MH (Metaheuristics)

1. Introduction

This work introduces the Root Sequence Index (RSI) allocation problem, provides four mathematical formulations, and details several solution approaches. The RSI is a parameter used in telecommunications, specifically in the random access procedure for establishing upload channels between 5G users and antennas or radios. If two or more neighbor radios have the same RSI number, then a collision can occur. When a collision happens, the user equipment may not identify each base station to connect correctly. RSI collisions can cause an increase in connection failures [Veríssimo et al., 2018].

This problem gains importance due to the massive growth in connection density expected with the advent of the fifth generation of wireless communication systems [Liu et al., 2018]. In the context of massive networks, the allocation of parameters to radio or base stations gain several levels of complexity.

We may define the RSI allocation problem as follows: let $G = (V, E)$ be an undirected graph, in which V is the set of vertices (radios) and E the set of neighborhood relations. For each vertex $v \in V$, we have to attribute an integer $L_{min} \leq w_v \leq L_{max}$ so that, for each pair of vertices $(uv) \in E$, $D_{min} \leq |w_v - w_u| \leq D_{max}$. We assume that each radio v has an old configuration O_v, which indicates an initial allocation that may be illegal or infeasible. The value of w_v may be interpreted as the RSI of the antenna v, or as the color of the vertex v.

Figure 1 represents how a network can be configured, in which the minimum distance between the RSI of neighbor antennas is ten units and there is no maximum distance. Note that, although there are only adjacent antennas inside a hexagon in this figure, the signals overlap between antennas in real life. Hence the problem cannot be modeled as a simple planar graph.

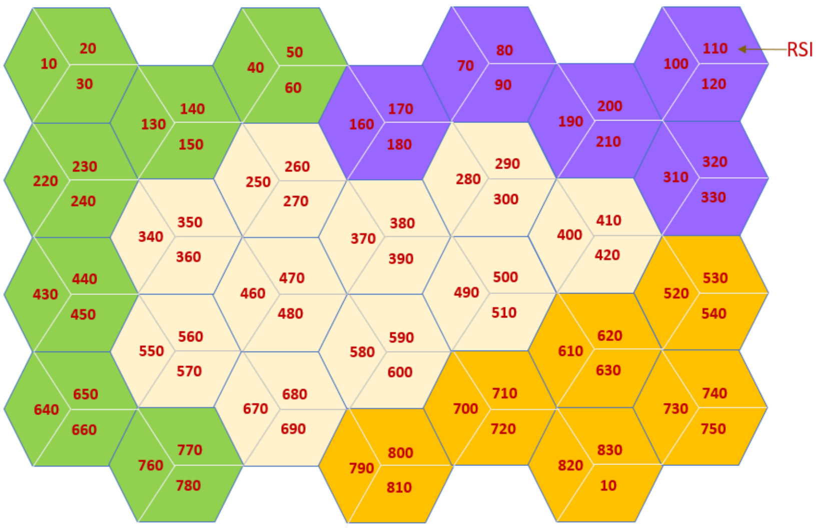

Figura 1: Representation of RSI allocation. Each hexagon represents a radio base station with three antennas in coverage angles of 120º each. The edges of each hexagon indicate adjacent antennas, which must have at least a difference of ten units between their RSI. Hexagons of different colors indicate towers in different regions or markets.

In practice, radio access network (RAN) engineers have two distinct operation scenarios for RSI assignment, and therefore we propose two distinct objective functions, one for each case.

In the first scenario, new nodes are deployed or removed from the network. In this case, we have deep changes in the network topology, and we want to guarantee that, for future node additions, we will have available RSI ranges for assignment, even though we may need to change the RSI for the most of the network (or slice of the network being considered in the optimization). In such a case, we try to maximize the smallest difference s_{min} between the RSI of neighbors. We must consider such need for the expected ultra-density in these networks and the expected increase in the coverage both in urban and rural areas.

In the second scenario, the number of nodes does not change, but their neighborhood relations do. Such a situation usually happens when we have a reconfiguration in the network where pilot power, frequencies, slices, and other RAN parameters change for some reason. Such changes induce new neighborhoods, i.e., adding or removing edges in the neighborhood graph. Such a situation is far more frequent than in the first scenario, and therefore, we want to avoid changes as mush as possible. In this scenario, we want to minimize the number of changes $\sum_{v \in V} c_v$ between new and old network configurations, to avoid disruptions and degradation in the service quality. To clarify what a expected result should be, Figure 2 portrays the optimal solution for an instance with 30 antennas in this scenario.

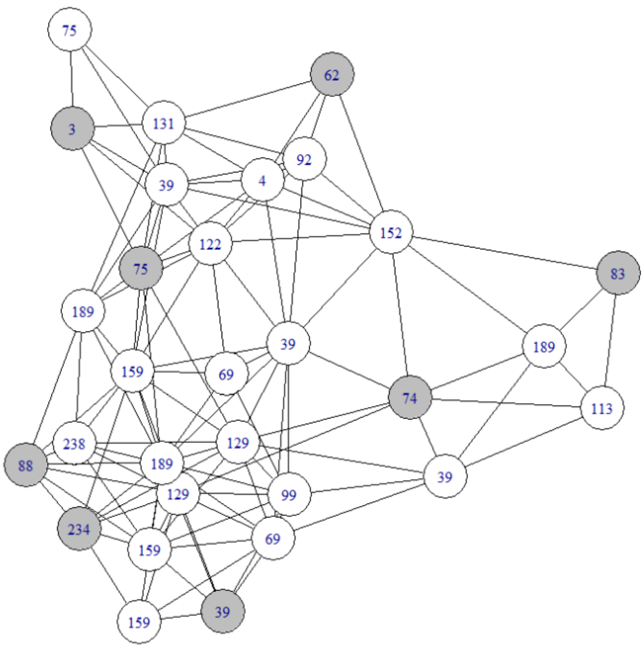

Figura 2: Optimal solution for instance `long_n0030_r030_150`. RSI values are indicated on each node, and a gray coloring implies that the configuration of the vertex was not changed.

1.1. Contributions and results

Our work introduces the RSI allocation problem, a novel problem in the context of 5G networks. Due to space constraints, the results are presented on a summarized manner. The reader should refer to the full dissertation for details [Londe, 2021].

Several publications resulted from this dissertation. Among them, there is a patent filled on the United States Trademark and Patent Office (USPTO) and a journal article currently in revision, both based on the full results of this study. Preliminary results of the mathematical models were presented on a national conference paper and its respective presentation, and an extended abstract on an international conference.

Patent: Andrade, C.E., Londe, M. A., Pessoa, L.S., Shankaranarayanan, N.K., Stawiarski, S. Facilitating Assignment Of Root Sequence Indexes While Minimizing Network Changes. The United States Patent and Trademark Office, filled. US Patent Application No. 17/540,880. Deposit date: 02/Dec/2021.

Journal publication: Londe, M.A., Andrade, C.E., Pessoa, L.S. (2022) Exact and heuristic approaches for the root sequence index allocation problem. Submitted to Applied Soft Computing, currently in R2.

Conference paper: Londe, M. A., Pessoa, L. S., Andrade, C. E. (2020). Modelos exatos para alocação do Root Sequence Index/ Exact Models for allocation of the Root Sequence Index. In: Anais do LII Simpósio Brasileiro de Pesquisa Operacional. Campinas: Galoá. https://proceedings.science/sbpo-2020/papers/modelos-exatos-para-alocacao-do-root-sequence-index.

Extended abstract: Londe, M. A.; Hokama, P. H. B.; Pessoa, L. ; Andrade, C. E. (2020). Exact And Heuristic Methods For The RSI Allocation Problem. INFORMS 2020. Available in https://www.abstractsonline.com/pp8/#!/9022/presentation/11323.

2. Formulations and definitions

We developed four models for the RSI allocation problem, based on the scenarios described before. The difference between the four models are the presence of non-linear constraints, and the objective functions and the constraints related to their calculation. Several parameters are considered on the problem formulation. Among them, the possible L_{min} and L_{max} values depend on the cell size and its frequency, and are classified in two sequences. The long-sequence is usually used in macrocells, with $L_{min} = 0$ and $L_{max} = 839$. The short-sequence is generally used in small cells with $L_{min} = 0$ and $L_{max} = 139$.

The D_{min} and D_{max} values are derived from network configuration. D_{min} is computed with a constant related to the division of frequency spectrum in the network (which has the value of 64 for 5G networks), the maximum value of the sequence L_{max}, and the cyclic shift value N_{CS}. The cyclic shift also originates from network characteristics, therefore, the minimum required distance is the same for all radios in the same network and sequence, as they have the same values of L_{max} and N_{CS}. For the D_{max} value, one must consider that the 5G network is inserted in the context of ultra-dense networks. Ultra-dense networks result from capacity enhancements needed for 5G and are synonymous with antennas or radios stationed only a few meters away from each other, thus potentially interfering with each other [Hao et al., 2016]. This indicates that defining a maximum distance D_{max} between the difference of the RSI of neighbors is adequate. Such a constraint allows setting a higher number of antennas in each network so that to improve coverage and network access.

Currently, there are no existing works that treat RSI allocation with the optimization focus, as far as the authors are aware. However, the proposed mathematical models consider characteristics of a modified Graph Coloring Problem (GCP) [Cormen et al., 2009], the Frequency Assignment Problem (FAP) [Hale, 1980], and the T-Coloring Problem (TCP) [Liu, 1992]. Those three problems have been proved as NP-Hard [Junosza-Szaniawski e Rzkażewski, 2014], and tend to be solved using heuristic methods. In fact, genetic algorithms are one of the most frequently used algorithms for graph coloring and correlated problems and show varying levels of success, especially in combination with other heuristics in hybrid genetic algorithms.

3. Solving the RSI allocation problem

Due to the difficulty of the models to find good or even feasible solutions, this work uses the Multi-Parent Biased Random-Key Genetic Algorithm with Implicit Path-Relinking (BRKGA-MP-IPR) introduced by Andrade et al. [2021], a multi-parent variant of the standard BRKGA. BRKGA has had many successful applications in telecommunications and network problems [Resende, 2012]. BRKGA-MP-IPR was specifically chosen because a multi-parent alternative can have better results than the classical one for genetic approaches to graph coloring [Lü e Hao, 2010].

Six decoders were customized for this problem, each using different strategies. These decoders use different ways of interpreting the chromosome, or even different chromosome characteristics, to obtain a possible coloring for the graph. All decoders presented here permit illegal colorings, i.e. colorings in which the minimum-maximum differences are not respected in one or more edges. These conflicting edges are penalized in the objective function by a factor C. However, a correction procedure is applied in the end, trying to solve these constraint violations. A local search procedure may also be performed during the decoding process.

The *Logic Direct* decoder (LD) is named as such due to its use of logical variables for the relation between the values of the RSI of neighbor nodes. These boolean variables indicate whether a vertex can or cannot be colored with a given RSI. This decoder interprets the chromosome as the order in which vertices are colored, being thus $|V|$-sized, with $|V|$ indicating the number of vertices in the instance. Similar to decoder LD, the *Logic Indirect* decoder (LI) also considers the logical constraints. However, in this variant, indirect neighbors (i.e., neighbors of neighbors) are also taken into consideration. This is the only difference between these two logic decoders. Again, the chromosome is interpreted as the order in which vertices are colored, being $|V|$-sized. The *Simple Coloring* decoder (SC) also uses a $|V|$-sized chromosome, but instead of ordering the vertices, this decoder directly uses each gene's value. The possible RSI values are normalized so that each gene corresponds directly to an RSI, which is then allocated to its vertex. This decoder has the characteristic of generating many infeasible solutions compared to its counterparts due to the strong random component in its generation.

The *Ordered Restricted Coloring* decoder (OR) uses a $2|V|$-sized chromosome. In this case, genes between $[0, |V| - 1]$ are used to extract an ordering of vertices and genes $[|V|, 2|V| - 1]$ correspond to a position inside the set of allowed RSI of a vertex. These allowed RSI are obtained by the neighborhood relationships between colored and uncolored neighbors of a vertex, and they are updated each time a vertex is colored. Differing from the previous decoders, the *Color Ordered by Degrees* decoder (KD) uses a $k + 1$-sized chromosome, with k indicating the number of possible colors. This decoder gives an ordering of colors. The vertices are ordered by decreasing values of degrees. Therefore, the algorithm prioritizes to color vertices with the significant potential to create more conflicts (i.e., more difficult vertices to color) first. Similar to the previous decoder, the *Color and Vertex Ordered* decoder (KC) uses both orders of colors and vertices to obtain a coloring for a graph. However, instead of obtaining the order of vertex from the network configuration, this decoder obtains it from the chromosome. The first part of the chromosome $[0, k + 1]$ has the order of the colors, while the second part $[k + 2, k + |V| + 1]$ becomes the order of the vertices.

The corrective procedure only happens if there are conflicts, i.e., the coloring is illegal for a particular solution. This procedure is similar to the classical Breadth First Search (BFS) algorithm for graphs. The procedure visits the adjacent nodes of all already visited vertices and changes the coloring of these neighbors if the edge has a conflict. All vertices are visited, and coming back to an already visited vertex is not allowed. The correction procedure ends when all vertices are visited. This procedure effectively finds feasible solutions from non-feasible ones, but it may still result in non-feasible solutions.

The local search procedure occurs at the end of every decoding and tries to find better solutions in the 1-color exchange space. That is, the procedure searches better solutions that are obtained by changing the color of one vertex of the analyzed initial solution. The local search is based on either Best Improvement (BI) or First Improvement (FI) strategies. In addition, local search BI may be performed on only part of the solution space, defined by a percentile LS% of the available vertices in each instance. On minimize-changes objective, the local search procedure changes the new color of a vertex to its old coloring and then recalculates the related coloring cost. If a vertex already has its old configuration, then the procedure skips it. The maximize-minimum-edge-span objective has a different procedure. For each vertex, it identifies its smallest legal edge. Then, the adjacent vertex related to this edge has its color changed to the best available value, which is the one that increases the most the edge value without violating the maximum possible difference of the instance. In the case of the vertex having no legal edges, then it is skipped by the procedure.

Classical BRKGA starts by creating the population with randomly generated individuals. Nonetheless, recent works [Martarelli e Nagano, 2020; Chagas et al., 2021] have shown that the introduction of good solutions to the initial population increases the algorithm's performance. The idea is to create one or several initial good solutions – generally based on a greedy algorithm – and introduce them to the initially generated population as an individual. For the RSI allocation problem, the warm start procedure colors the vertices with the smallest possible color, considering the minimum and maximum difference between the RSI of the vertices. This procedure colors the vertices in non-increasing degree order, i.e., the vertices with higher degrees are colored first. Note that the solution generated by this procedure may be infeasible. This procedure uses the initial coloration for both objectives. The initial coloration is more important for the minimize-changes objective, but it is also considered for the maximize-minimum-edge-span case.

4. Experiments and results

The developed methods were applied on 139 novel instances based on real scenarios of a global telecommunications company. They are divided into two groups which indicate whether the RSI is in the long or short sequence, with 82 long sequence instances and 57 short sequence instances. Each instance has a number of vertices, the minimum and maximum distances, the old configuration of each node, and the existing edges. The experiments were conducted on a cluster of identical machines, each with processor Intel Xeon E5-2650 CPU 2.0 GHz (12 cores / 24 threads) and 128 GB of RAM, running CentOS Linux 6.9. For the mathematical models, solver IBM ILOG CPLEX 12.10 was used, while BRKGA-MP-IPR was programmed on C++, both on a maximum of one hour, four threads, and 100 GB of RAM. As BRKGA-MP-IPR has a higher amount of parameters for tuning than classical BRKGA, the application of design of experiments for tuning each parameter is too complex and time-consuming. Therefore, the *iterated racing* [López-Ibáñez et al., 2016] procedure was used to perform the parameter tuning.

Regarding the mathematical models, one may note that the linear models had a smaller number of instances with no results for both types of sequences and objectives functions, in comparison with the non-linear models. Furthermore, those instances solved by the linear models have higher numbers of vertices, which indicates more difficulty and a greater number of variables. In addition, for all models, the number of unsolved long sequence instances is higher than the amount for the short sequence for both objective functions. Again, this is indicative of the impact of more variables and restrictions for the long sequence in comparison with the short sequence, pointing out to a higher level of complexity.

Regarding the BRKGA decoders, first, in relation to the number of conflicts, the behavior of the decoders in relation to long and short sequences is similar in both objectives. In both cases, the short sequence has a higher percentage of infeasible runs, but a much lower amount of conflicts in those infeasible runs. Meanwhile, for a same objective, the amounts of runs feasible and infeasible are close for instances on both sized-sequence groups. This indicates that the objective function is more influential in the hardness of converging to feasible solutions than the instances' characteristics. In general, the change from short to long sequence means an increase in conflict amount for decoders OR and SC, and a reduction in all other decoders. This is true for both objective functions. About the corrective procedure, it appears to not have a significant impact on solution quality – except for decoder SC in the long sequence. In both objectives, decoder SCno has a much smaller number of average conflicts than its counterpart. This can be explained by the fact that decoder SC procedure has a severe increase in time requirements because of the correction procedure. In this way, the SCno manages to run more generations and thus converge to better solutions.

Now, going to special remarks about specific decoders, decoder KC has a strange performance in both cases. It has a very high percentage of infeasible solutions, and the feasible solutions tend to reach the optimal costs. This is especially true for the short sequence cases, as all or almost all feasible solutions are optimal. That can be caused by one of two alternatives: first, the decoder will only converge to feasible solutions by happenstance, and always will hit the best solution. This option could also explain why the time to reach optimal solutions, in this decoder (as shown in the performance profile plots) is always low. Another reason may be that this decoder needs more time to converge to feasible solutions when it does not in the starting minutes. That could explain the high amount of infeasible solutions and the good performance in the feasible ones, and also be related to the fact that decoder KC uses a $n + k + 1$-sized chromosome, which would be negatively affected by any increase in size and possible color amount of the instances. Meanwhile, the performance of decoder OR has poor performance in all four cases. Exceptionally in the long sequence-maximizing minimal span case, decoder OR has surprisingly good statistics, comparable to decoder LD, but still holds its place as the worst decoder. Finally, among the best decoders are LD and LI. Both manage to converge to feasible solutions quickly and have the best performances in all cases - with LD being still better than LI in all cases.

5. Conclusion

This work has several contributions. We introduce the RSI allocation problem, whose main application is for 4G and 5G radio access networks. We propose four mathematical formulations for it, and observe the effect of linear and non-linear constraints in model efficiency. A biased random-key genetic algorithm was customized for this problem, including six decoding approaches, two local searches, and a correction procedure. The strategy of considering vertex order and neighbor relations was shown to be efficient in finding viable, good solutions in tractable computational times. It should be highlighted that this is the first study of RSI allocation with an optimization focus.

For future works, one must consider the development of a stochastic version of the RSI allocation problem. The allocation of 5G parameters is rife with uncertainty, as the existence and relative location of antennas changes day-to-day. Another possibility is the study of similar alternatives to decoder LD, as this procedure showed itself to be effective and thus should be explored in similar problems. Finally, one may consider developing coloring strategies that do not use the old configuration for the maximize-minimum-edge-span objective.

Referências

Andrade, C. E., Toso, R. F., Goncalves, J. F., e Resende, M. G. C. (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 03772217 (ISSN).

Chagas, J. B. C., Blank, J., Wagner, M., Souza, M. J. F., e Deb, K. (2021). A non-dominated sorting based customized random-key genetic algorithm for the bi-objective traveling thief problem. *Journal of Heuristics*, 27(3):267–301. ISSN 13811231 (ISSN).

Cormen, T. H., Leiserson, C. E., Rivest, R. L., e Stein, C. (2009). *Introduction to algorithms*. MIT press.

Hale, W. K. (1980). Frequency assignment: Theory and applications. *Proceedings of the IEEE*, 68 (12):1497–1514.

Hao, P., Yan, X., Ruyue, Y.-N., e Yuan, Y. (2016). Ultra dense network: Challenges enabling technologies and new trends. *China Communications*, 13(2):30–40.

Junosza-Szaniawski, K. e Rzkażewski, P. (2014). An exact algorithm for the generalized list t-coloring problem. *Discrete Mathematics and Theoretical Computer Science*, 16(3):77–95.

Liu, D. D.-F. (1992). T-colorings of graphs. *Discrete Mathematics*, 101(1-3):203–212.

Liu, X., Wu, S., Guo, Y., e Chen, C. (2018). The demand and development of internet of things for 5G:a survey. In *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, p. 1–2.

Londe, M. A. (2021). The RSI allocation problem: exact and heuristic methods. Master's thesis, PUC-Rio. Available in https://doi.org/10.17771/PUCRio.acad.53566.

López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., e Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58. ISSN 2214-7160.

Lü, Z. e Hao, J.-K. (2010). A memetic algorithm for graph coloring. *European Journal of Operational Research*, 203(1):241–250.

Martarelli, N. J. e Nagano, M. S. (2020). Unsupervised feature selection based on bio-inspired approaches. *Swarm and Evolutionary Computation*, 52. ISSN 22106502 (ISSN).

Resende, M. G. (2012). Biased random-key genetic algorithms with applications in telecommunications. *Top*, 20(1):130–153.

Veríssimo, R., Vieira, P., Rodrigues, A., e Queluz, M. P. (2018). PCI and RSI conflict detection in a real LTE network using supervised learning. *URSI Radio Science Bulletin*, 364:11–19.