

Implementation of the Metaheuristic GRASP Applied to the School Bus Routing Problem

Wilson Soares de Siqueira^{1,2*}, Fernando Jorge Ebrahim Lima e Silva^{1,2}, Elvis Nascimento da Silva^{1,2}, Raimunda Vieira Santos da Silva¹, Marcelo Lisboa Rocha²

¹Systems Development Department, Federal Institute of Tocantins, Brazil.

²Computer Science Department, Federal University of Tocantins, Palmas, Brazil.

* Corresponding author. Tel.: +55(63)8474-3380; email: wilsonsoares@ifto.edu.br

Manuscript submitted October 9, 2015; accepted February 18, 2016.

doi: 10.17706/ijeec.2016.6.2.137-145

Abstract: The school bus problem routing (SBRP) is an important practical problem studied in combinatorial optimization of operational research. It is formulated through a set of stops, buses, schools and garage, where from these sets, we seek to create optimized routes to reduce the operating cost of the service. This paper presents a solution to the SBRP, using the GRASP applied to a real problem. This meta-heuristic is divided into two stages: the construction of a viable solution and followed by a Local Search procedure. These two phases are repeated in each iteration. In the construction phase, a greedy and random function is used to construct an initial solution. It is also presented compared to the widely used heuristic and good results in the literature.

Key words: Combinatorial optimization, school bus, GRASP, SBRP, metaheuristic.

1. Introduction

The school bus routing problem (SBRP) is a problem of combinatorial optimization, commonly referred to in operational research. Through a literature review, which found the number of publications available in leading journals, note that the problem is little explored in the literature, when compared to the number of jobs that treats the classic problem routing vehicles (VRP).

In the literature, it was found SBRP case studies in Brazil. Ref. [1] it presents a proposed heuristic method to solve the problem and a case study in the state of Parana, Brazil and presented the resolution for rural and urban areas. Ref. [2] it presents a case study of SBRP to rural areas in the state of Minas Gerais, Brazil. Ref. [3] it presents a solution for urban transportation to the city of Curitiba, Parana, Brazil. Ref. [4] it presents a methodology location of schools and optimizing the school transport in rural areas. This methodology is based on two models of mixed integer programming. The first deals with the school's location and sizing issues, and the second deals with the school bus routing. The case study was conducted in the city of Grajau Baron state of Maranhao in Basil.

The objective of this work is to create a method to solve the SPREP, considering the service in rural and urban areas, considering time windows, as well as a fleet of heterogeneous vehicles.

2. Description the School Bus Routing Problem

The School Bus Routing Problem (SBRP) is a combinatorial optimization problem which was modeled by

Newton and Thomas [5]. The SBRP is a variation of the Vehicle Routing Problem (VRP) [6], which is commonly treated in operations research.

The SBRP aims to optimize the school bus transport, where the student is pickup in a bus stop located in front of his residence or close to it, then the student is transported and delivered in your school at the end of the school day the student is transported again for same location where it was pickup.

In the literature can be found several problems, methods and constraints to solve the SBRP. Ref. [7] it points out that the start of school time takes place at the same time, therefore, it generates a large bus demand at the beginning and end of the school day.

Ref. [8] it shows in his work on the size of the SBRP objectives of the proposal and propose a multi-objective solution, to minimize the number of buses used for pickup and delivery of students and improve the quality of the service offered. For this, treated separately each objective, considering not only the optimal solution, but also the set of measured way.

Ref. [9] it presents a model for SBRP taking into consideration the attendance to students with specific needs or vulnerable because of health problems. Ref. [10] presents an application for monitoring of school transport in Europe to transport vulnerable users (children, people with specific needs or poor health).

In the work [11] emphasizes on investment of the Ministry of Education of China, which invested in the launch of a pilot program of school bus optimization, which were chosen in 2011, six cities to perform the tests in a real environment, so search for the best practices for school bus service.

Ref. [12] developed a randomized algorithm Location Based Heuristics (LBH) and showed a linear regression method for time duration of the pickup of students and for the duration of delivery time from at school, it was also considered the bus waiting time. To solve the problem it was used the Mix Load method. An improvement to the algorithm is presented by [13]. The LBH algorithm was implemented in this work, to carry out computational tests and compare with proposed results of this paper.

2.1. The Metaheuristic GRASP

The Greedy Randomized Adaptive Search Procedure (GRASP) It was developed by [14]. It is an iterative procedure where each iteration consists of a construction phase an initial solution Greedy randomized adaptive way and then a heuristic is applied for improvement, typically a local search procedure. The best solution among all the iterations is stored [14]. The Fig. 1 shows the pseudocode GRASP.

2.2. The Constructive Algorithm

The proposal is to develop a system decision support for SBRP, for it will be used the metaheuristic GRASP. In the literature are found resolution methods and formulation so that the problem is treatise separately, such as the types of fleet (homogeneous or heterogeneous) and types of attendance (in areas rural or urban). The aim is to create an algorithm to solve the general form problem.

Considering that, each set corresponds to a table in one database and that has attributes, primary keys and foreign keys. The following steps are presented for the construction method: In step 1, must be selected a parameter (Urban or Rural), respectively represented by binary numbers (0 and 1). In step 2, according to the last parameter, the sets are filtered from the Database, and the sets are loaded into Memory RAM. In the step 3 is started the execution of the construction method.

Given the following set of data entries, where P is the set Stops, O is the set Bus, G is the set of Garage, E is the set Schools and R is the set of routes, initiated empty. In phase 1, the loop is started the constructive method, a conditional is then checked if the solution set $R' = \emptyset$ or the bus hours of attendance $>$ that start time of school is randomly selected an o the set O , is taken at random. If the bus hours of attendance $<$ school start time, the same bus will be used to serve a new route. The bus has a capacity constraint, where is not permitted to their limit is exceeded. When the bus hours of attendance constraint is satisfied, the

buses used will be taken from the set O .

In the phase 2, is executed the function $RCLi(\alpha, ge, P)$ it is responsible for calculating and creating an array of costs from an output location of the bus, which may be the garage G or the school E , until all the stops $p \in P$, (see in Fig. 2), that calls the function $createRCL(arrayCosts)$ which uses criteria of GRASP, to create a set of best stops $p \in P$, (see in Fig. 4). All details are shown in the session 2.3.

In the phase 3, $ci \leftarrow RCLi(\alpha, ge, P)$, it is chosen at random a stop $pi \subset ci \mid ci \in P$, then, the solution R' get the stop pi and the $school(i)$ which is associated with pi , starting a route Rn , after being inserted into R' , the stop pi is removed the set P .

In the phase 4, While the bus capacity \leq the quantity pickup students, a new bus stop is visited. An iterative procedure is carried out by the function $cj = RCLj(\alpha, school(i), pi, P)$, a manner similar to phase 3. Therefore, the following parameters are sent: the stop pi and the $school(i)$, after, the function $RCLj(\alpha, school(i), pi, P)$ it is responsible for calculating and creating an array of costs between the stop pi until all the stops $p \in P$ that has as its destination the $school(i)$, (see in Fig. 3) that calls the function $createLCR(arrayCosts)$ which uses criteria of GRASP, to create a set of best stops $p \in P$, (see in Fig. 4). All details are shown in the session 2.3.

Algorithm 1. Constructive Method for the Metaheuristic GRASP

```

1:  $P = \{1, 2, \dots, p\}$  // Set the stops not visited
2:  $O = \{1, 2, \dots, o\}$  // Set the buses
3:  $G = \{1, 2, \dots, g\}$  // Set the garage
4:  $E = \{1, 2, \dots, e\}$  // Set the schools
5:  $R = \{\emptyset\}$  // Set empty, for gets the routes
6: WHILE ( $P \neq \emptyset$ ) DO{
7:    $b =$  a randomly bus selected the set  $O \in G$ ;
8:    $ci = RCLi(\alpha, ge, P)$ ; // Restricted Candidate List (RCL), all stops
9:    $pi =$  a randomly bus selected the set  $LCRi$  de  $ci \in P$ 
10:   $R' = pi, school(i)$ ; // school  $e$  associated with stop  $i$ 
11:   $P = P$ ; // removed stop  $pi$  the set  $P$ ;
12:  WHILE ( $capacityBus \leq quantityStudents$ ) DO {
13:     $cj = RCLj(\alpha, school(i), pi, P)$ ; //RLC, algorithm GRASP
14:     $pj =$  a randomly bus selected the set  $cj \in P$ ;
15:    SE ( $arrivalTime \leq timeStartingSchool$ ){
16:       $R' = j, school(i)$ ;
17:       $P = P$  // removed stop  $j$  the set  $P$ ;
18:    }
19:    ELSE {
20:      Break executing and the bus goes to  $school(i)$ ;
21:    }
22:  }
23:   $R = R + R'$ 
24: }
25: Output the Solution  $R$ 

```

Fig. 1. Constructive method for the SBRP with GRASP.

In the phase 5, $cj \leftarrow RLCi(\alpha, school(i), g, P)$, where is selected at random a stop $pj \subset cj \mid cj \in P$, after, the solution R' get the stop pj e $school(i)$ which is associated with pj , therefore, a new stop pj is inserted in the route Rn . After being included in R' , the stop pj is removed the stop P , provided that it meets the restrictions of phase 6.

In the phase 6, if the bus arrival time at school \leq start time of the school day. At each iteration are added the following attributes in the solution R' , the duration of the the route time of a stop i until j (time in seconds), the quantity pickup students, the duration of time to pickup all the students in the stop p , (time in seconds) and the landing time students at school. The iterative method continues to until the bus capacity

constraint or restriction beginning the school day, are not violated. If the arrival time at school or the capacity of the bus, satisfy the constraints, the bus goes destined for school and left the students.

In the phase 7, is used the algorithm 2-opt, to improve the solution R' . The details are shown in section 2.4.

In the phase 8, Shows the solution of all the built routes, $R = \{R1, R2, \dots, Rn\}$

2.3. Restrict Candidate List

Accordance with the initial parameters of the functions $RCLi(\alpha, g, P)$ e $RCLj(\alpha, school(i), g, P)$, an array of costs is constructed. The algorithm 2, which is the function $RCLi(\alpha, ge, P)$, creates an array of insertion cost for the construction of a route R' as of P . Where are calculated the cost of the garage path g or school e until each stop p and the stop p until the school e associated with stop p , so, $Cost = Ci + Cj + Ck$. (one example is showed in the Table 2).

Algorithm 2. Restricted Candidate List Initial (RCLi) - Considers all the stops

```

1: FUNCTION RCLi ( $\alpha, ge, P$ ){
2:     WHILE ( $P \neq \emptyset$ ) DO {
3:          $pi = P[n]$ ;
4:          $e = school(i) \subset P[n] \in E$ ;
5:          $cost = distance(gi, pi) + distance(pi, e)$ ;
6:          $arrayCosts = arrayCosts + cost$ ;
7:     }
8:      $createRCL(arrayCosts)$ ;
9:     RETURN RCL;
10: }
```

Fig. 2. Array the distance the RCLi.

Algorithm 3. Restricted Candidate List Initial (RCLj) – Considers only the stop the $school(i)$

```

1: FUNÇÃO RCLj ( $\alpha, school(i), pi, P$ ) {
2:     WHILE ( $P \neq \emptyset$ ) DO {
3:          $pj = P[n]$ ;
4:         IF ( $pj \in school e$ ) THEN {
5:              $cost = distance(pi, pj) + distance(pj, e)$ ;
6:         }
7:          $arrayCosts = arrayCosts + distance$ ;
8:     }
9:      $createRCL(arrayCosts)$ ;
10:    RETURN RCL;
11: }
```

Fig. 3. Array the distance the RCLj.

The algorithm 3, is the function $LCRj(\alpha, school(i), pi, P)$, It has similar characteristics to algorithm 2, therefore, is insertet a the paremeter $school(i)$, which is intended to filter only the stops p , having as destination the $school(i)$. The array cost is calculated from the stop pi , which was defined in (line 9) of the algorithm 1.

After the creation of the array cost the function $createRCL(arraycosts)$ is called to identify the best potential stops p , to start construction the R' . In the construction phase generates a viable solution to the problem through the method partially greedy and partially random. Where the parameter α has the function to determine the size of the Restricted Candidate List (RCL). The α is a value ranging from 0 until 1, where $\alpha=0$, determining that the greedy method is totally and $\alpha=1$, determines that the method will be totally random. This parameter is responsible for determining the size of the CRL, the nearest to 1, the value of α , the greater the number of stops, added together and the worse the solution to the problem.

For this study was considered the $\alpha=0, 1$. (One example is showed in Table 2). At each iteration of

Algorithm 1 is built a RCL, which is formed from the array cost, where the best potential stops are selected, with the formula $GRASP = arrayCosts p \in P \leq min + \alpha*(max - min)$ showed in (line 6) the algorithm 4, being realized a grouping in a subset of potential viable stops, after, a stop is randomly selected, taken from the RCL subset, that has the best stops candidates. The Fig. 4, show the algorithm for RCL.

Algorithm 4. GRASP: Greedy Randomized Adaptative Search Procedure

```

1: FUNCTION createRCL(arrayCosts){
2:     WHILE (arrayCosts ≠ ∅) DO {
3:         checkCosts = arrayCosts[n];
4:         min = min (arrayCosts p ∈ P);
5:         max = max (arrayCosts p ∈ P);
6:         GRASP = arrayCosts p ∈ P ≤ min + α*(max - min);
7:     IF (GRASP ≤ checkCosts) THEN {
8:         RCL = GRASP;
9:     }
10:    RCL = RCL + RCL;
11: }
12: RETURN RCL;
13: }
```

Fig. 4. Greedy randomized adaptative search procedure.

2.4. Improvement for Initial Solution with Algorithm 2-opt

The improvement heuristic 2-Opt was proposed by Croes [15], the idea of this improvement heuristic is quite simple given a set R' that a route built in R_n , eliminate two edges of the solution and reinsert two edges cross, therefore, if the edges were removed peers that connect the stops $(k1, k2)$ and $(j1, j2)$, the inserted edges connect the stops in the following way $(k1, j2)$ and $(j1, k2)$. If this new configuration is better than the previous, so, if the distance reduce, keep the new route. Otherwise, choose again two edges for analysis.

3. GRASP Application to SBRP

The Fig. 5, show one example, where, is considered one Garage, five stops an one school. It is also present the distances between each stop. To demonstrate the functioning of the GRASP applied to SBRP. To begin construction of a school bus route, is considered that the bus moves garage and should visit all the stops to collect the students and let them in school.

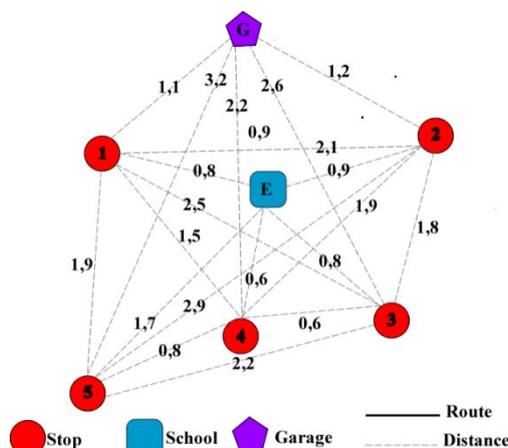


Fig. 5. Problem representation to be solved.

In iteration 1, (showed in Table 1) calculates the distance between the garage until each stop, therefore,

creating an array cost. Applying adaptive parameter $\alpha=0, 1$, the following formula:

$$GRASP = p(P) / arrayCosts p \in P \leq min + \alpha*(max - min)$$

It obtained the following result to GRASP = 2.2, how can be seen in iteration the Table 2. Then, All the stops with lower distance 2.2, will be inserted in the set RCL = {P1 and P2}, After, one stops is randomly selected. After, is raffled P2. See that, P1 and P2 are the closest stops from Garage.

In iteration 2 (showed in Fig. 1) is calculated the distance between P2 until each stop unvisited. After, the resulted GRASP = 2.71, and the stops P3 and P4 They are inserted into the RCL and is raffled P3.

The procedure is carried until all the stops are visited.

Table 1. Criteria Used for Selection of the Stops with GRASP, Considering $\alpha = 0.1$

Array the Costs = $(C_i, C_j) + (C_j, C_k)$						
	P1	P2	P3	P4	P5	Randomly
Iteration 1: G → P _n → E	1,9	2,1	3,4	2,8	4,9	P2
Iteration 2: P2 → P _n → E	2,9	-	2,6	2,5	4,6	P3
Iteration 3: P3 → P _n → E	3,3	2,7	-	1,2	3,9	P4
Iteration 4: P4 → P _n → E	2,1	2,8	1,4	-	2,5	P1
Iteration 5: P1 → P _n → E	-	3,0	3,3	2,1	3,6	P5

legend stops unavailable Stops selected Stops unvisited

Table 2. Results the Selection of the Stops with GRASP, Considering $\alpha = 0.1$

$GRASP = p(P) / arrayCosts p \in P \leq min + \alpha*(max - min);$	Stops the RCL	Randomly
Iteration 1: $1,9 + 0,1*(4,9-1,9) = 2,2$	{ P1 e P2 } ≤ 2,2	P2
Iteration 2: $2,5 + 0,1*(4,6-2,5) = 2,71$	{ P3 e P4 } ≤ 2,71	P3
Iteration 3: $1,2 + 0,1*(3,9-1,2) = 1,47$	{ P4 } ≤ 1,47	P4
Iteration 4: $2,1 + 0,1*(2,5-2,1) = 2,14$	{ P1 } ≤ 2,14	P1
Iteration 5: $3,6 + 0,1*(3,6-3,6) = 3,6$	{ P5 } ≤ 3,6	P5
Solutio Initial	R = {G, P2, P3, P4, P1, P5, E, G}	
After improvement with 2-opt	R = {G, P2, P3, P4, P5, P1, E, G}	

After the iterative procedure, the following solution was generated for R = {G, P2, P3, P4, P1, P5, E, G}. As can be seen in Fig. 6, the route built is not optimized, for improving the route was used heuristic algorithm 2-opt, to leave the optimal route. The Fig. 7 shows the optimized solution that is represents by R = {G, P2, P3, P4, P5, P1, E, G}.

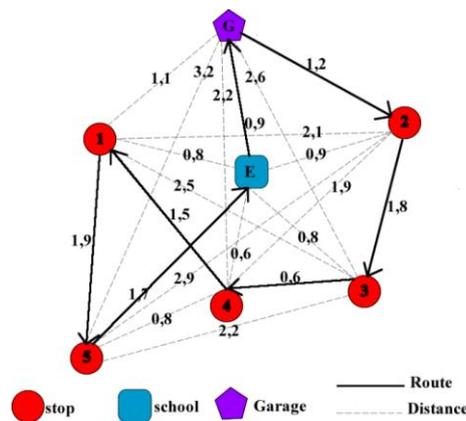


Fig. 6. Generated route by GRASP.

To compare the computational results were implemented: the algorithm LBH the Braca [12], which is

much cited in the literature for solving the problem and the metaheuristic GRASP [14].

For this problem, it was considered a total of 87 stops, 8 schools, and 1169 students. the results are showed in Table 3.

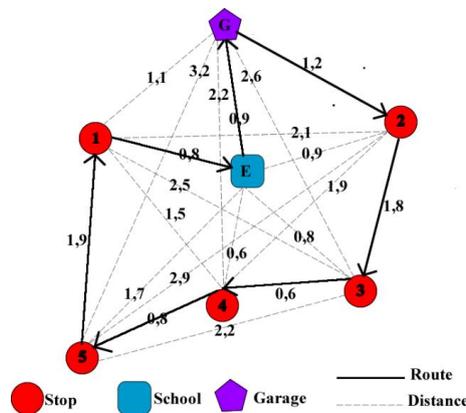


Fig. 7. Route with improved algorithm 2-opt.

To obtain the results shown in table 03, have been performed 20 iterations for algorithm LBH and GRASP, where have been performed the average bus used, computational time and the cost total, which is the sum of the distance of all routes in kilometers (KM). The computational results showed a savings of 36,29% in cost total in KM, an economy the 32,11% in quantity the bus necessary and 54,17% reduction in run time, with the purpose of this paper.

Table 3. Results Computational Using 20 Iterations

Algorithm LBH		Algorithm GRASP	
Necessary bus	7.2	Necessary bus	5.45
Computational time	0.0463309 (s)	Computational time	0.0300501 (s)
Total cost (KM)	111.70	Total cost (KM)	81.953

4. Conclusion

With the utilization of GRASP is obtained good results in comparison with the heuristic LBH. Have been performed 20 iterations for algorithm. The computational results showed GRASP of improvements over the LBH. An economy the 36,29% in cost total, an economy the 32,11% quantity the bus necessary and 54,17% reduction in run time.

5. Future Works

In future work, we intend to develop a decision support system for SBRP, based on the implementation of meta-heuristic algorithm GRASP.

References

[1] Souza, L., & Siqueira, P. (2010). Heuristic methods applied to the optimization school bus transportation routes: A real case. *Trends Appl. Intell. Syst.*

[2] Faraj, M. F., Sarubbi, J. F. M., et al. (2013). Estudo de caso: O problema do transporte escolar rural em minas gerais. *Simpósio Bras. Pesqui. Operacional*, 1332–1343.

[3] Steiner, M., & Zamboni, L. (2000). O problema de roteamento no transporte escolar. *Pesqui. Operacional*.

[4] Mandujano, P., Giesen, R., & Ferrer, J.-C. (2012). Model for optimization of locations of schools and

student transportation in rural areas. *Transp. Res. Rec. J. Transp. Res. Board*, 2283, 74–80.

- [5] Newton, R. M., & Thomas, W. H. (1969). Design of school bus routes by computer. *Socioecon. Plann. Sci.*, 3(1), 75–85.
- [6] Dantzig, G., & Ramser, J. (1959). The truck dispatching problem. *Manage. Sci.*
- [7] Fügenschuh, A., Martin, A., & Stöveken, P. (2005). Integrated optimization of school starting times and public bus services. *Oper. Res. Proc.*
- [8] Corberán, R., Fernández, A., Laguna, E., & Martí, M. (2000). Heuristic solutions to the problem of routing school buses with multiple objectives. *J. Oper. Res. Soc.*, 53.
- [9] Kamali, B. S. J. M., & Pohl, E. A. An analysis of special needs student busing. *J. PUBLIC Transp.*, 16.
- [10] Kalogirou, K., Chalkia, E., Bekiaris, E., & Diederichs, F. (2012). An application for the information of children according their school transportation. *Procedia-Social and Behavioral Sciences*, 48, 363–372.
- [11] Chen, X., Kong, Y., Dang, L., Hou, Y., & Ye, X. (2015). Exact and metaheuristic approaches for a bi-objective school bus scheduling problem. *PLoS One*, 10(7), p. e0132600.
- [12] Braca, J., Bramel, J., Posner, B., & Simchi-Levi, D. (1997). A computerized approach to the New York City school bus routing problem. *IIE Trans.*, 28(8), 693–702.
- [13] Park, J., Tae, H., & Kim, B. I. (2012). A post-improvement procedure for the mixed load school bus routing problem. *Eur. J. Oper. Res.*
- [14] Feo, T., & Resende, M. (1995). Greedy randomized adaptive search procedures. *J. Glob. Optim.*
- [15] Croes, G. (1958). A method for solving traveling-salesman problems. *Oper. Res.*



Wilson Soares de Siqueira was born in Ceres, Goias in 1981. He graduated in information systems the State University of Goias (2008). He was specialized in computer networks for Ananguera Faculty of Anapolis (2010). He was a master student in computational modeling and systems at the Federal University of Tocantins (2013). He is currently a professor of computer science in Federal Institute of Tocantins, Campus Araguatins, Brazil and computing project area coordinator, the institutionally scholarship program for initiation to teaching, the Federal Institute of Science and Technology Tocantins, Brazil.



Fernando Jorge Ebrahim Lima e Silva was born in Recife, Pernambuco in 1975. He graduated in information systems in the Uninorte (2000). He was specialized in computers network at the Federal University of Rio Grande do Norte (2003). He was a master student in computational modeling and systems at the Federal University of Tocantins (2013). He is currently a professor of computer science in Federal Institute of Tocantins, Campus Palmas, Brazil.



Elvis Nascimento da Silva graduated in technology in the data processing foundation University of Tocantins (2001), MBA in information technology management for Tocantinense Institute of Postgraduate, ITOP (2008) and specialization in the database by the Catholic University of Tocantins FACT (2009). He is currently a master in computer modelling systems, Federal University of Tocantins, UFT. He is professor of basic education, technical and technological of the Federal Institute of Education, Science and Technology of Tocantins IFTO. It has experience in computer science, with emphasis on Information Technology and Database.



Raimunda Vieira Santos da Silva was born in Tocantinópolis, To, Brazil in 1965. She got a master's degree in production engineering and specialized in audit and government management from PUC-GO (2014). She graduated in business administration from Federal University of Pará (2004). It has experience in strategic and environmental management; bidding; audit and public finance. Currently she is working on the environmental agenda and strategic planning at the Federal Institute of Education,

Science and Technology of Tocantins.



Marcelo Lisboa Rocha was born in Rio de Janeiro in 1972. She is graduated in computer science from the Catholic University of Petropolis (1994), the master in computer Science from Federal University Fluminense (1997), the master in electrical engineering from the Federal University of Rio de Janeiro (1999) and the doctorate in electrical engineering from the Federal University of Rio Janeiro (2008). He is currently reviewer the periodic INFOCOMP Journal of Computer Science and associate Teacher 4 of the Federal University

of Tocantins, Brazil. It has experience in computer science, acting on the following topics: metaheuristics, combinatorial optimization, mathematical programming, computer networks and high-performance computing.