Journal of Engineering Science and Technology Review 14 (6) (2021) 220 - 227

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

Research Article

Genetic Algorithm Applied to the Capacitated Vehicle Routing Problem: A Study on the Behavior of the Population of Genetic Algorithms Considering Different Encoding Schemes and Configurations of Genetic Operators

Stanley Jefferson de Araujo Lima and Sidnei Alves de Araújo*

Informatics and Knowledge Management Postgraduate Program, Nove de Julho University – UNINOVE, Rua Vergueiro, 235/249 – Liberdade, Zip Code: 01504-001, São Paulo/SP, Brazil

Received 29 July 2020; Accepted 21 October 2021

Abstract

The application of Genetic Algorithm (GA) in solving any combinatorial problem presupposes the adoption of an encoding scheme and configuration of genetic operators that, according to the literature, impact the behavior of the GA population during the convergence phase. Understanding this behavior is essential to assist the refinement of configuration parameters and for proposing heuristics that support searching better quality solutions with the least possible computational effort. However, observing and understanding such behavior is not an easy task and, for this reason, this issue has attracted the attention of many researchers in recent years. In this work we proposed a computational tool and a method to evaluate the impact of different encoding schemes and settings for crossover and mutation operators in the GA performance. To this end, we have considered the application of GA in solving the Capacitated Vehicle Routing Problem (CVRP). However, it is important to highlight that the computational tool and the evaluation method are generalizable for the study of other population-based meta-heuristics and/or other combinatorial optimization problems. The results indicate that in most aspects binary encoding schemes are less efficient than schemes using integer numbers, and that the impact caused by genetic operators is directly related to the employed encoding scheme. It was also found that some of the performance measures proposed can be used either to propose heuristics or as heuristics itself.

Keywords: Genetic Algorithm, Capacitated Vehicle Routing Problem, Encoding Scheme, Genetic Operators, Population Behavior.

1. Introduction

The Genetic Algorithm (GA) is a metaheuristic method, derived from evolutionary computation that is based on Charles Darwin's theory of natural evolution, according to which organisms in a population that best adapt to the environment in which they live are more likely to survive and reproduce. The evolution process of the GA occurs from a population of initial individuals, through the application of genetic operators for selection, crossing and mutation of individuals [1]. Each individual of the population (or chromosome) is a possible encoded solution to the addressed problem.

The GA has been widely used in the solution of highly complex optimization problems, known in the literature as NP-Hard (Non-deterministic Polynomial Time), among which are the Job Shop Scheduling Problem (JSSP), Cutting and Packing Problem (CPP) and the Vehicle Routing Problem (VRP). The latter, addressed in this work, has attracted the attention of researchers in recent years due not only to the difficulty of its solution, but also to its presence in various practical situations [2].

According to Vieira [3], the VRP consists in determining a set of routes to be followed by a fleet of vehicles, so that the demands of all customers are satisfied, and that each vehicle returns to the depot at the end of the route. The objective can be, for example, minimize total cost, travel time or total travel distance. In the literature there are many variations of VRP, such as Capacited Vehicle Routing Problem (CVRP), Vehicle Routing Problem with Pickup and Delivery (VRPPD), Vehicle Routing Problem with Time Window (VRPTW), among others. The CVRP, treated in this work, is the most common version of the VRP and considers a single fleet for a transport that leaves and returns to the unique depot. The restriction is limited to the vehicle's capacity [2,3].

The methods applied to solve routing problems can be exact, heuristic or metaheuristic and are chosen, in most cases, according to the size and complexity of the solution space [4]. The GA has been widely used in solving the VRP and other optimization problems due to the good results presented in the literature [5,6]. However, solving any optimization problem using GA presupposes the adoption of a solution encoding scheme (computational representation of the solution in the GA chromosome) and the configuration of genetic operators, which, according to the literature, directly impact the behavior of the population in the solution space.

Regarding the use of GA to solve the CVRP there are many works in the literature of the last decade such as [7-22]. However, only few works such as Lu & Vincent [9], Ruiz et al. [20], Hosseinabadi et al. [21], Zhu [22] and Koç et al. [23] have investigated the mechanisms of GA functioning.

Lu & Vincent [9] combined different operators and parameters of the GA to solve the VRPPD with flexible time windows. In this study, the authors explored the combination of different operators to maximize the quality of the solution found. However, they do not describe how these operators

*E-mail address: saraujo@uni9.pro.br
ISSN: 1791-2377
doi:10.25103/jestr.146.25

impact the behavior of the GA population when exploring the solution space.

Ruiz et al. [20] employed the GA with random keys to solve the CVRP. For this purpose, a real-number vector encoding scheme was used and, for the solutions refinement, the authors applied local search heuristics to support the GA in the exploration of the solution space helping it to escape from local optima. Although the authors addressed elements about the functioning of the GA, they do not explain how the refinement heuristics employed impact the behavior of the GA.

Hosseinabadi et al. [21] and Zhu [22] presented hybrid versions of GA for CVRP solution. Hosseinabadi et al. [21] combined GA with the Gravitational Emulation Local Search – GELS Algorithm [24]. In the proposed approach, the GELS algorithm is used as a heuristic to refine GA solutions. According to the authors, the crossover and mutation operators are not enough to guarantee that the GA adequately explores the solution space, and for this reason they employed the GELS algorithm to strengthen the solution space exploration process. They also argue that the combination of these two algorithms produces a variety of solutions that lead to the exploration of a large area in the solution space, making the GA more efficient to escape local optima, thus obtaining solutions with better quality. Zhu [22] proposes an approach combining GA and Fuzzy C-Means Clustering techniques that modifies the crossover and mutation operations, so that these operators can be set dynamically. According to the author, the improved algorithm achieved good performance reducing the possibility of falling into local minima in the search process. However, both works [21, 22] do not present detailed explanations on how the proposed hybrid GA explores the solution space.

Koç et al. [23] presented a hybrid GA for the optimization of the VRPTW. In this proposal, the authors combined GA with the Large Neighborhood Search (LNS) heuristic However, although they have explored concepts of intensification and diversification, there are no discussions on the behavior of the GA population in the spaces of solutions produced.

As can be seen, most works in the literature reporting the use of GA to solve the VRP have focused on the hybridization of this metaheuristic and not on understanding the mechanisms that control its functioning or that explain how the GA population is impacted by the genetics operators and the solution encoding schemes. On the other hand, in recent years there has been increasing research efforts focused on mitigating the solution space in order to understand the difficulties of metaheuristics in solving NP-Hard problems, as well as exploring this knowledge to develop more efficient methods [25]. In this sense, some authors employ the idea of fitness landscape, which consists of forming a surface that represents the space of solutions [26].

The technique used by some researchers involves generating several solutions and plotting the cost of these solutions versus the distance from the solution to the best-known solution, to visualize the fitness landscape. Nevertheless, to explore the fitness landscape, other researchers employ projection methods to generate the position of each element in a two or three-dimensional space, such as the t-Distributed Stochastic Neighbor Embedding (t-SNE), Fastmap, Isomap, Least Square Projection (LSP) and Principal Component Analysis (PCA) [27, 28]. The latter was used by Tayarani-n & Prügel-bennett [29] to study fitness landscape in the Traveling Salesman Problem (TSP).

Despite advances in mapping and exploring spaces of solutions, understanding the behavior of populations of metaheuristic is not an easy task, mainly due to the high dimensionality of such spaces and the lack of specific tools for this task. It is in this context that the present work is inserted presenting as main contributions a tool and an evaluation method that allow investigating the influence of the encoding scheme and genetic operators on the behavior of the GA population and, consequently, on the quality of the solutions provided by it. For that, experiments were carried out considering the GA in the CVRP solution.

2. Theoretical Background

2.1 Genetic Algorithm

The Genetic Algorithm (GA) is an optimization method based on the evolutionary process, that is, it is based on Charles Darwin's theory of evolution of species [1]. In general, the operation of the GA consists of maintaining a population of individuals (chromosomes), initially generated randomly, representing possible solutions to a given problem, which evolve over generations (iterations) through a process of competition in which the best solutions (defined according to their aptitudes) are more likely to survive and reproduce. The reproduction is based on a process of selecting and modifying candidate solutions. For that, genetic operators such as selection, crossover and mutation are used [1, 2]. The operation of the GA is illustrated in Figure 1.

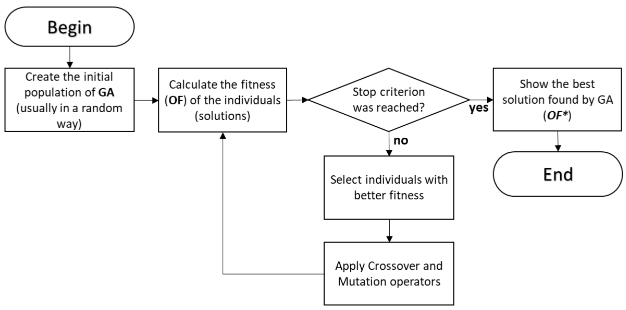

Fig. 1. GA operation

A chromosome is constituted by genes, which in turn represent the decision variables of a problem [2,14]. Thus, encoding a solution means defining an architecture that encodes the information of the problem addressed, to allow the computational interpretation of the variables that define its solution [30], as illustrated in Figure 2.

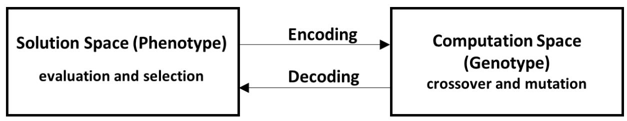

Fig. 2. Encoding and decoding operations

Although there are many operators for manipulating the individuals of the GA population, the main ones are: selection, crossover and mutation. The selection operator aims to choose the individuals from the population that should participate in the reproduction process, passing on their characteristics to the next generation. The selection is based on the fitness of each chromosome, which is calculated by an objective function (OF). The main selection methods are roulette wheel, ranking and tournament [2]. The crossover is responsible for carrying out the exchange of genetic material between pairs of individuals (parents), that is, it consists of the recombination of genes from selected individuals to generate offspring. Finally, the mutation operator, normally applied after crossover, aims to increase the diversity and variability of the population, helping to avoid premature convergence of the GA (fall into local minima). Mutation can occur in two ways: by changing gene positions or by replacing allele values, as shown in Figure 3.

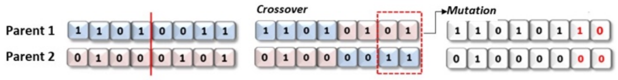

Fig. 3. Crossover and mutation operations

The performance of the GA can be influenced by the definition of the parameters used in the configuration of its operators [31, 32]. However, it is very difficult to define a set of optimal values for such parameters that guarantee a better performance of the GA, since the same configuration can present variation of results when applied in different encoding schemes and problems [32]. Nevertheless, the solution encoding scheme is also of great importance for the good performance of the GA, since it is directly related to the quality of the solutions found, as well as the computational cost spent to find them [30].

The main encoding schemes employ vectors and matrices of binary, integer and reals numbers; symbols; vector of characters and tree structures [33]. According to Castro [34], there is no encoding scheme would work equally well in all situations. Thus, in each case a careful choice must be made aiming to obtain the expected result.

2.2 Capacitated Vehicle Routing Problem (CVRP)

The CVRP is the most basic version of VRP in which all customers have a demand defined previously and must be fully met by only one vehicle, the fleet is homogeneous (all vehicles are similar in terms of capacity) and depart from a single distribution center (depot). In this version of the problem, only the vehicle capacity constraint is imposed so that the sum of the demand of all customers on a route cannot exceed the vehicle capacity [35]. Figure 4 illustrates an example of a CVRP, in which 3 routes are defined to meet the demands of 10 spatially dispersed customers.

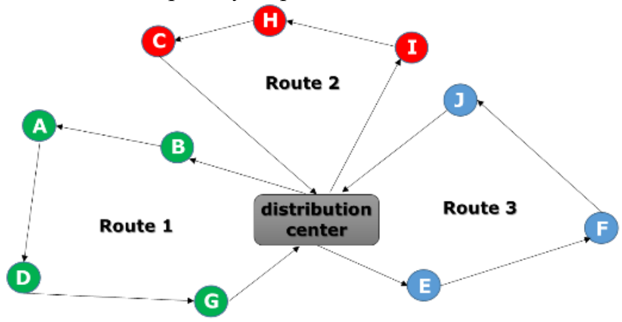

Fig. 4. Example of a CVRP with 3 vehicles and 10 costumers

Mathematical formalizations for the CVRP, taking into account its intrinsic constraints, can be found in [2, 3, 14]. Such formalizations are used as objective function (FO) to evaluate the aptitude (fitness) of the solutions generated by the employed optimization method.

3. Material and Methods

3.1 Instances of CVRP

To carry out the computational experiments, we considered three instances (collection of data describing CVRP scenarios) from the set proposed by Christofides [36], which are detailed in Table 1. The optimal solutions (best solutions) for these instances were extracted from the work of Reinelt & Wenger [37].

Table 1. Characteristics of the scenarios described by the considered instances of CVRP

	Instances		
	E-n22-k4	E-n51-k5	E-n76-k8
Vehicle capacity	6000	140	180
Costumers and their demands	1(0), 2(1100), 3(700),4(800), 5(1400), 6(2100), 7(400), 8(800), 9(100), 10(500), 11(600), 12(1200), 13(1300), 14(1300), 15(300), 16(900), 17(2100), 18(1000), 19(900), 20(2500), 21(1800), 22(700)	1(0), 2(7), 3(30), 16), 5(9), 6(21), 7(15), 8(19), 9(23), 10(11), 11(5), 12(19), 13(29), 14(23), 15(21), 16(10), 17(15), 18(3), 19(413), 20(9), 21(28), 22(8), 23(8), 24(16), 25(10), 26(28), 27(7), 28(15), 29(14), 30(6), 31(19), 32(11), 33(12), 34(23), 35(26), 36(17), 37(6), 38(9), 39(15), 40(14), 41(7), 42(27), 43(13), 44(11), 45(16), 46(10), 47(5), 48(25), 49(17), 50(18), 51(10)	1(0), 2(18), 3(26), 4(11), 5(30), 6(21), 7(19), 8(15), 9(16), 10(29), 11(26), 12(37), 13(16), 14(12), 15(31), 16(8), 17(19), 18(20), 19(13), 20(15), 21(22), 22(28), 23(12), 24(6), 25(27), 26(14), 27(18), 28(17), 29(29), 30(13), 31(22), 32(25), 33(28), 34(27), 35(19), 36(10), 37(12), 38(14), 39(24), 40(16), 41(33), 42(15), 43(11), 44(18), 45(17), 46(21), 47(27), 48(19), 49(20), 50(5), 51(22), 52(12), 53(19), 54(22), 55(16), 56(7), 57(26), 58(14), 59(21), 60(24), 61(13), 62(15), 63(18), 64(11), 65(28), 66(9), 67(37), 68(30), 69(10), 70(8), 71(11), 72(3), 73(1), 74(6), 75(10), 76(20)
Optimal solution	375	521	735

Such instances were classified by their degree of optimization difficulty, as presented in the work of Kalatzantonakis et al. [38] and by its size taking into account the number of customers ($n-1$) and vehicles (k) indicated in their names. For example, the E-n22-k4 instance considers a CVRP scenario with 21 customers plus the depot and 4 vehicles. According to Kalatzantonakis et al. [38], it is classified as low difficulty and small (few vehicles and few customers). The instance E-n51-k5 was classified as medium difficulty and size (medium number of vehicles and/or customers) and, finally, the E-n76-k8 was classified as high difficulty and large size (many vehicles and/or customers).

3.2 Encoding Schemes
From the literature review, the most representative encoding schemes were identified and two of them were chosen: binary (three-dimensional matrix of binary numbers), since it is more natural for the GA and for its wide use in the literature for solving combinatorial optimization problems, including VRP; and integer (vector of integers), as it is the most recurrent in recent literature. The chosen encoding schemes were presented in Nazif & Lee [8] and in Lima et al. [14] and detailed below.

3.2.1 Binary Encoding Scheme (BES)
The CVRP solution encoding scheme illustrated in Figure 5 was employed by Lima et al. [14] and by Vieira [3]. It is a three-dimensional matrix of binary numbers with *M* columns, *N* rows and depth *Z*, where *M* represents the number of customers to be served, *N* represents the customer service sequence and *Z* is defined by the number of vehicles (routes) needed to meet the total demand. The column position that receives value 1 in each row indicates the customer to be visited while the row position indicates the customer's service order. The depth represents the vehicle assigned to serve a given customer. This type of encoding, although more natural for the GA, requires more computational instructions to decode the solution.

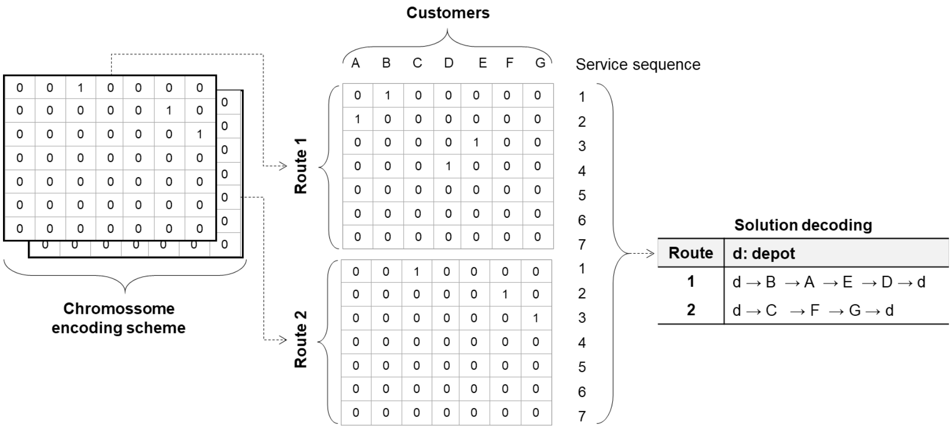

Fig. 5. Operation of considered BES. Source: adapted from Lima et al. [14]

3.2.2 Integer Encoding Scheme (IES)
Nazif & Lee [8] employed a vector of integers of length *N* (see Figure 6), where *N* represents the number of customers to be served, and each element of the vector can contain an integer value corresponding to a customer. The sequence of numbers in the vector determines the customer service order and the set of customers that make up each route is delimited by the vehicle's capacity. As illustrated in Figure 6, when the vehicle's capacity is exceeded, a new route is started. In Hosseinabadi et al. [21] a similar representation was used, with the only difference that each route is separated by a digit 0. In this kind of encoding, the vector represents the chromosome and each element of the vector represents a gene.

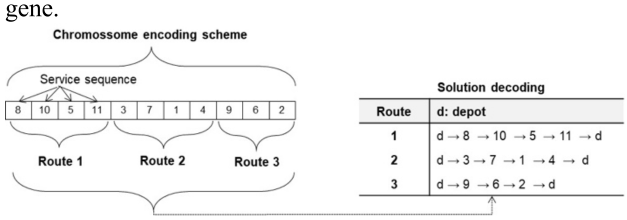

Fig. 6. Functioning of considered IES. Source: adapted from Nazif & Lee [8]

3.3 Parameters and Genetic Operators Configurations
In order to find out which genetic operators can most significantly impact the behavior of the GA, as well as define the rate intervals (configuration) to be used in these operators, preliminary experiments were carried out taking into account crossover rate, number of crossover points, mutation rate, elitism rate, population size and number of generations.

From these experiments it was observed that crossover and mutation operators have more significant impact on the behavior of the GA and therefore they appear in the literature among the main configuration parameters. In addition, as in Abdelatti et al. [10], the Design of Experiments (DoE) was employed to define the set of GA configuration parameters presented in Table 2.

Table 2. GA configuration parameters

Parameters	Adopted values/rates
Population size (*popSize*)	200
Number of generations (*nGer*)	50
Number of crossover points (*nPoints*)	1
Crossover rate (*cr*)	(0.50, 0.70, 0.90)
Mutation rate (*mr*)	(0.01, 0.05, 0.10)
Elitism rate (*er*)	0.10
Method of selection (met*Sel*)	roulette wheel

3.4 Experimental Design
To carry out the computational experiments, the GA was configured with the parameters presented in Table 2 and applied in the resolution of the three instances described in Table 1, considering BES and IES.

Then, using the developed tool (see section 4.1), the results of the experiments were analyzed, based on the following measures GAP, percentage of non-feasible solutions (PNFS), diversity (DIV), dispersion (DISP), area explored (AEX) and computational cost (CC), in order to understand the behavior of the GA population. To make a fair comparison, we considered the same initial population (generated randomly with uniform distribution) in all experiments presented. In addition, two individuals generated by Gillett & Miller heuristic [39] were inserted in the initial population, allowing the GA starting with two feasible solutions, as demonstrated in [14, 17, 18].

4. Results

4.1 Developed Computational Tool

The computational tool employed in the experiments carried out in this work, illustrated in Figure 7, was developed to visualize and analyze the behavior of the GA population in the solution space. Each individual of the population is projected by the PCA method from a *n*-dimensional solution space to a two-dimensional space, represented by the panel at the right of the interface shown in Figure 7, composed by 720×640 pixels representing the points of the projected solution space.

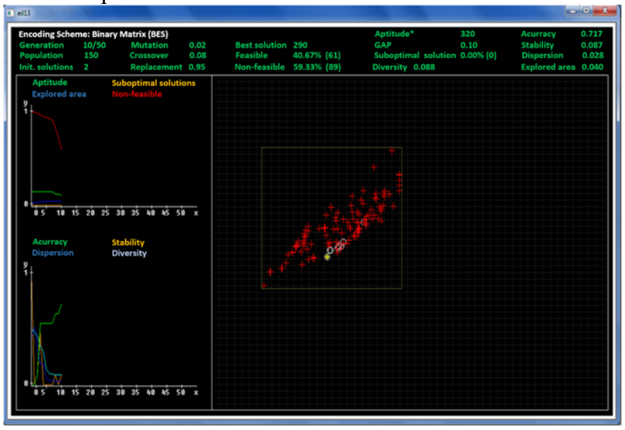

Fig. 7. Computational tool interface

There are two important characteristics in the two-dimensional space projected by the PCA: i) two similar solutions in the phenotype space are always mapped to neighboring points in the projected space and; ii) it is possible that two very similar solutions in the genotype space are mapped to the same point in the projected space since the dimensionality reduction is very drastic.

Regarding the layout of the computational tool, the left panel displays two graphs plotted at runtime. The first presents the measure of aptitude or fitness, the explored area and the percentages of feasible and non-feasible solutions. The second graph illustrates the accuracy, stability, dispersion and diversity of the population. Such measures are presented further in this section.

The tool also has a status bar, located at the top, which displays the parameters employed by GA such as: encoding scheme, population size, number of generations, current generation, number of feasible individuals inserted in the initial population, crossover rate, mutation rate, elitism and the objective function value of the best solution found in the literature for the instance ("Best solution" corresponds to *OF*_best used to calculate the GAP). Still in the status bar are shown the computational cost (time in seconds between the beginning and the end of the GA execution in the optimization of an instance) and the values calculated during the GA execution for 8 different measures, listed below, being the first 4 extracted from the literature and the last 4 proposed in this work.

- **Accuracy (ACC):** aims to determine the location of the best solution found, within the range defined by a lower limit representing the worst solution found and an upper limit, indicated by the best solution known in the literature [40].
- **Diversity of the population (DIV):** expresses, through a value ranging from 0 to 1, the diversity of the population. Value 0 indicates that all individuals in the population are similar while value 1 indicates that all individuals are completely different.

2. **Stability (STA):** indicates how much the algorithm is able to maintain its stability. A stable algorithm maintains accuracy for countless generations [40].

- **GAP:** expresses how far the result obtained for a problem, denoted by OF (objective function value), is from the best result reported in the literature for that problem, represented by (OF_{best}). It is calculated as follows: GAP = (*OF* - *OF*_best) / *OF*_best.
- **Area explored (AEX):** reflects, through a rate ranging from 0 to 1, the percentage of the area of the solution space is explored during the execution of the GA. The higher the rate, the greater the area explored, indicating that the algorithm was more likely to explore more promising points. To compute this measure, the two-dimensional projected solution space is divided into 2,500 subregions (50 columns × 50 rows). Then, the number of subregions explored by one or more individuals mapped by PCA over all generations is divided by the total number of subregions.
- **Dispersion (DISP):** describes how dispersed the individuals of the population are, that is, the greater the average dispersion, the greater the area explored in the solution space. For that, a distance matrix is calculated, and the average distance value is normalized to the interval [0, 1].
- **Percentage of feasible solutions (PSF):** expressed through a rate ranging from 0 to 1, it reflects the capacity of the GA in converting non-feasible solutions in feasible ones. The **Percentage of non-feasible solutions (PNSF)** is obtained simply making 1- PSF.
- **Percentage of suboptimal solutions (PSS):** this measure reveals, through a rate, the percentage of suboptimal solutions in the population. Here, suboptimal solutions are those with $GAP \leq 0{,}10$.

The results generated by the tool are recorded in a text file in CSV (Comma Separated Values) format that can be read in Microsoft Excel spreadsheet software, and in video using AVI (Audio Video Interleave) format, facilitating an in-depth and detailed analysis of the populations' behavior.

Finally, it is noteworthy that although the computational tool provides several performance measures, some of them describe the behavior of the GA in a similar way. Thus, we decided to consider only the GAP, DIV, AEX and computational cost.

4.2 Analysis of the Behavior of the GA Population

This section presents a general analysis of the behavior of the GA population, based on the comparison of measures that sharply highlight the differences observed in the use of the two encoding schemes and the configuration parameters presented in sections 3.2 and 3.3 respectively. The results obtained in the experiments are summarized in Table 3, which contemplates the average values for the performance measures considered. The values highlighted in blue indicate the best performances of the GA while the values in red indicate its worst performances. The graphs of figures 8 to 11 were provided to help understanding these results.

In relation to the average GAP, it is observed in the graph of Figure 8 that, in general, in all the experiments, the IES presented GAPs smaller than those obtained by BES. In this context, employing IES, the experiment using (cr = 0.7 and mr = 0.01) resulted in the best average GAP (0.22). Nevertheless, using BES the best average GAP (0.37) was obtained in the experiment adopting (cr = 0.9 and mr = 0.01).

Table 3. Summarized results

GA configuration parameters		BES			IES		
cr	mr	GAP	DIV	AEX	GAP	DIV	AEX
0.5	0.01	0.43	0.28	0.09	0.23	0.70	0.33
0.5	0.05	0.48	0.28	0.11	0.24	0.73	0.34
0.5	0.10	0.52	0.29	0.11	0.24	0.69	0.35
0.7	0.01	0.38	0.28	0.12	0.22	0.71	0.33
0.7	0.05	0.44	0.29	0.13	0.24	0.74	0.35
0.7	0.10	0.45	0.29	0.13	0.23	0.70	0.38
0.9	0.01	0.37	0.24	0.08	0.25	0.71	0.30
0.9	0.05	0.43	0.24	0.09	0.26	0.73	0.31
0.9	0.10	0.45	0.24	0.09	0.25	0.69	0.34

On the other hand, the worst performance of the GA regarding the average GAP in the IES was obtained in the experiment using (cr = 0.9 and mr = 0.05), in which the value of 0.26 was obtained, while in the IES it was obtained the value 0.52 in the experiment adopting (cr = 0.5 and mr = 0.10).

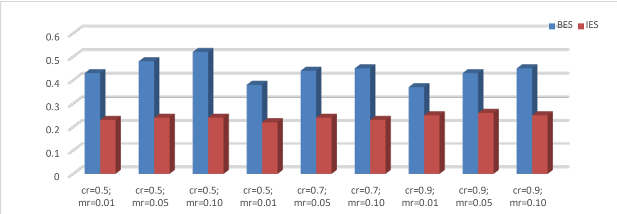

Fig. 8. Average GAP

The IES also favored the performance of the GA in terms of diversity (DIV), as shown in the graph in Figure 9, in which it is noted that in all experiments with the IES resulted in a greater population diversity than that obtained by using BES. IES provided the best performance of the GA in the experiment using (cr = 0.7 and mr = 0.05) reaching a maximum diversity of 0.74 and the worst performance in the experiments adopting (cr = 0.5 and mr = 0.10) and (cr = 0.9 and mr = 0.10) in which the maximum diversity was 0.69. BES presented the best performances in the experiments using (cr = 0.5 and mr = 0.10), (cr = 0.7 and mr = 0.05) and (cr = 0.7 and mr = 0.10) in which a maximum diversity of 0.29 was reached. The worst performances for BES were observed in the experiments adopting (cr = 0.9 and mr = 0.01), (cr = 0.9 and mr = 0.05) and (cr = 0.9 and mr = 0.10), where a maximum diversity of only 0.24 was obtained.

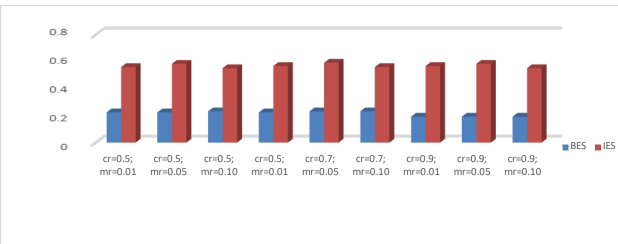

Fig. 9. Maximum diversity

About the average area explored (AEX), it is observed in the graph of Figure 10 that in this aspect the GA also obtained a better performance using IES, which presented its best performance adopting (cr = 0.7 and mr = 0.10), in which was reached the value of 0.38. On the other hand, in the experiment adopting (cr = 0.9 and mr = 0.01) IES presented the worst performance, obtaining 0.30 of average explored area. BES, in terms of this measure, presented the best performance in the experiments using (cr = 0.7 and mr = 0.05) and (cr = 0.7 and mr = 0.10), obtaining 0.13 of average explored area, and the worst performances in the experiment employing (cr = 0.9, mr = 0.01), in which 0.08 was obtained for the referred measure.

The low values presented in the explored area for the BES can be explained by the fact that it is a very sparse encoding scheme, which results in a very large space of solutions. On the other hand, IES, that is more compact and can be understood as a more direct representation of the solution, produces smaller spaces of solutions.

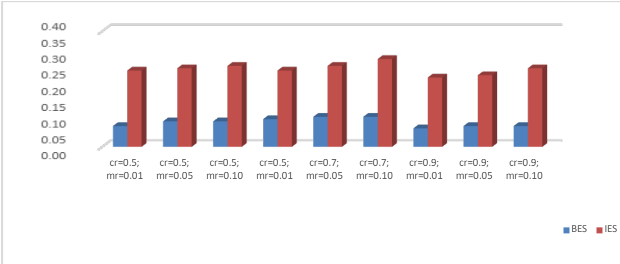

Fig. 10. Average area explored

Regarding the computational cost, the graph illustrated in Figure 11 shows the average processing time (in seconds) for the two encoding schemes, considering the three instances presented in section 3.2. From this graph one can observe that IES presented a computational cost equivalent to only 10.27% of the cost achieved by using BES. In this sense, although the BES presents greater simplicity in the operations performed because it is more "natural" for the GA, since it is sparse and requires more operations in the solution decoding, it demands a high computational cost.

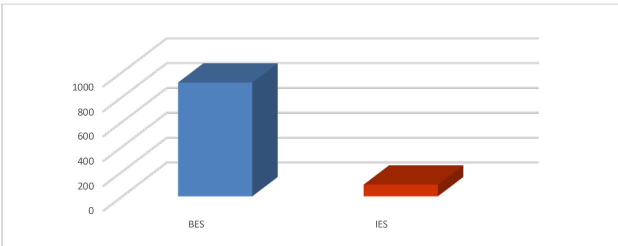

Fig. 11. Average computational cost

From the results achieved in the experiments, it is possible to infer that in the BES scheme the rates used in the crossover and mutation operators impact several aspects of the GA. It was observed that high crossover rates (for example, mr = 0.9) provide better GA performance in terms of convergence, however such rates imply a high number of non-feasible solutions. A medium crossover rate (e.g. cr = 0.7) provides a larger area explored in the solution space while low crossover rate (for example, cr = 0.5) produced a better performance of the GA in terms of the ability to transform non-feasible solutions into feasible ones, in increasing population diversity and population dispersion.

It was also observed that low mutation rates (e.g. mr = 0.01) provide a better performance of the GA in terms of convergence, ability to convert non-feasible solutions into feasible ones and population diversity. On the other hand, high mutation rates lead to a better performance of the GA in relation to population dispersion, as well as in the explored area.

In this context, it appears that it is possible to obtain good performance from the GA using BES, combining it with heuristics that reinforce the negative aspects caused by the configuration of genetic operators. For example, when employing a high crossover rate, it is suggested to adopt a low mutation rate and/or local search heuristics, with the objective of helping the GA in transforming non-feasible solutions into feasible ones, as well as heuristics that enable the GA to explore a larger number of regions, such as ILS (Iterated Local Search), (Simulated Annealing) and GRASP (Greedy Randomized Adaptive Search Procedure) [11]. On the other hand, when employing a low crossover rate, one should consider the use of heuristics to support the GA in the GAP evolution process, as well as heuristics that promote increased diversity and population dispersion.

When employing low mutation rates (for example, mr = 0.01) it is recommended to employ diversification heuristics with the purpose of increasing diversity, being an alternative to dynamically increase the population size, or even replace a part of the population by randomly generated individuals, with the purpose of increasing the diversity and dispersion of individuals in the population, promoting an exploration of the space of solutions with greater amplitude.

In the IES, the rates adopted in crossover and mutation operators subtly impact on some aspects of the GA. The results reveal that high crossover rates (e.g. cr = 0.9) help to maintain some population diversity and dispersion. On the other hand, a low crossover rate (cr = 0.5) favors convergence, as well as promoting a better performance of the GA in converting non-feasible solutions into feasible ones, in addition to increasing the distance between the solutions, resulting in a greater number of sub-regions explored in the solution space. However, a low crossover rate (e.g. cr = 0.5) provides a subtle improvement in convergence, in addition to resulting in the conversion of non-feasible solutions into feasible ones with fewer generations, as well as a greater distance between the solutions in the search space, resulting in a larger area explored.

In addition, when applying a high mutation rate, it is recommended to employ refinement heuristics that help the GA in the process of improving the GAP as well as improving its ability to convert non-feasible solutions into feasible ones, and, when adopting low mutation rates, it is suggested employ diversification heuristics that help the GA to carry out a more dispersed exploration through the solution space, in addition to maintain diversity and dispersion of the population at high levels.

To improve the GA performance using IES, it is recommended to combine crossover and mutation operators with additional heuristics that compensate the adverse aspects caused by the configuration adopted. It is also recommended to employ diversification heuristics that help the GA to make "jumps" in the solution space aiming to explore sub-regions that may be more promising.

Making a general analysis of the results obtained in the experiments, it can be seen that the crossover and mutation rates have a different impact on the GA performance depending on the solution encoding scheme adopted. The results also revealed that the encoding scheme adopted has a more significant impact on the behavior of the GA population than the crossover and mutation rates alone. These analyses corroborate the insights of the literature with regard to the need for great care in choosing of the solution encoding scheme as well as the configuration of the genetic operators, since an inadequate parameterization can significantly impair the performance of the GA in obtaining the expected results.

The results also showed that, even though GA is an effective, robust and flexible metaheuristic, it is recommended to combine it with other heuristics to solve NP-Hard problems, aiming to overcome its critical aspects. Thus, the discussions presented in this work can be of great importance to support research involving the application of GA in the solution of the CVRP and other VRP variants.

5. Conclusions

The results of the experiments carried out showed that crossover and mutation rates impact the GA differently depending on the solution encoding scheme adopted, and that the encoding scheme acts more significantly on GA behavior than crossover and mutation rates alone. The results also showed that the binary encoding scheme is less efficient in terms of converting non-feasible solutions into feasible ones, in addition to producing low population diversity, making the convergence of the GA difficult. In the other hand, integer encoding schemes are efficient in generating feasible solutions, as well as in providing good population diversity, thus helping the algorithm convergence process. Perhaps it is for this reason that this type of encoding has been widely used in current works, although this explanation is not provided in such works. Finally, in addition to the discussions on the analysis of the GA behavior, the performance measures provided here and incorporated in the developed computational tool can help in the proposition and/or choice of heuristics that aim to support the process of refining the solutions generated by the GA, improving its performance. In feature works we intend to analyze other encoding schemes, as well as other genetic operators; produce a set of rules to compose a Fuzzy inference mechanism responsible for the reconfiguration of the GA at runtime; and evaluate the performance of the GA using the mentioned Fuzzy inference mechanism.

Acknowledgments
The authors would like to thank UNINOVE for the scholarship provided to S. J. A. Lima. In addition, S. A. de Araújo would like to thank CNPq – Brazilian National Research Council for his research scholarship (Process 313765/2019-7).

References

1. Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press (1992).
2. Sbai, I., Limam, O., & Krichen, S. An effective genetic algorithm for solving the capacitated vehicle routing problem with two-dimensional loading constraint. International Journal of Computational Intelligence Studies, 9(1-2), p. 85-106 (2020).
3. Vieira, H. P. Metaheuristic for solving vehicle routing problems with time window. Dissertation (Master in Applied Mathematics) – Institute of Mathematics, Statistics and Scientific Computing, State University of Campinas, Campinas, Brazil (2013).
4. Benvenga, M. A., Araújo, S. A. D., Librantz, A. F., Santana, J. C., & Tambourgi, E. B. Application of simulated annealing in simulation

and optimization of drying process of Zea mays malt. Engenharia Agrícola, 31, p. 940-953 (2011).
5. Librantz, A. F. H., Coppini, N. L., Baptista, E. A., Alves de Araújo, S., & Castello Rosa, A. D. F. Genetic algorithm applied to investigate cutting process parameters influence on workpiece price formation. Materials and Manufacturing Processes, 26(3), p. 550-557 (2011).
6. Boukouvalas, D. T., Rosa, J. M., Belan, P. A., Tambourgi, E. B., Santana, J. C. C., & de Araújo, S. A. Optimization of cotton dyeing with reactive dyestuff using multiobjective evolutionary algorithms. Chemometrics and Intelligent Laboratory Systems, 219, 104441 (2021).
7. Gao, G. B., Zhang, G. J., Huang, G., Zhu, H. P., & Gu, P. H. Solving material distribution routing problem in mixed manufacturing systems with a hybrid multi-objective evolutionary algorithm. Journal of central south university, 19(2), p. 433-442 (2012).
8. Nazif, H., & Lee, L. S. Optimised crossover genetic algorithm for capacitated vehicle routing problem. Applied Mathematical Modelling, 36(5), p. 2110-2117 (2012).
9. Lu, C. C., & Vincent, F. Y. Data envelopment analysis for evaluating the efficiency of genetic algorithms on solving the vehicle routing problem with soft time windows. Computers & Industrial Engineering, 63(2), p. 520-529 (2012).
10. Abdelatti, M., Hendawi, A., & Sodhi, M. Optimizing a GPU-accelerated genetic algorithm for the vehicle routing problem. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, p. 117-118 (2021).
11. Sbai, I., Krichen, S., & Limam, O. Two meta-heuristics for solving the capacitated vehicle routing problem: the case of the Tunisian Post Office. Operational Research, p. 1-43 (2020).
12. Osaba, E., Onieva, E., Carballedo, R., Diaz, F., & Perallos, A. An adaptive multi-crossover population algorithm for solving routing problems. In Nature Inspired Cooperative Strategies for Optimization (NICSO 2013), p. 113-124. Springer, Cham (2014).
13. Ammi, M., & Chikhi, S. A generalized island model based on parallel and cooperating metaheuristics for effective large capacitated vehicle routing problem solving. Journal of computing and information technology, 23(2), p. 141-155 (2015).
14. Lima, S. J. A., Santos, R., & Araújo, S. Optimization of the capacitated vehicle routing problem using Genetic Algorithms and the Gillet and Miller and Down Hill heuristics. In National Meeting of Production Engineering, 35, pp. 1-15 (2015).
15. Sánchez-Oro, J., López-Sánchez, A. D., & Colmenar, J. M. A general variable neighborhood search for solving the multi-objective open vehicle routing problem. Journal of Heuristics, 26(3), p. 423-452 (2020).
16. da Costa, P. R. D. O., Mauceri, S., Carroll, P., & Pallonetto, F. A genetic algorithm for a green vehicle routing problem. Electronic notes in discrete mathematics, 64, p, 65-74 (2018).
17. Lima, S. J. D. A., & Araújo, S. A. D. A new binary encoding scheme in genetic algorithm for solving the capacitated vehicle routing problem. In International Conference on Bioinspired Methods and Their Applications, p. 174-184. Springer, Cham (2018).
18. de Araujo Lima, S. J., de Araújo, S. A., & Schimit, P. H. T. A hybrid approach based on genetic algorithm and nearest neighbor heuristic for solving the capacitated vehicle routing problem. Acta Scientiarum. Technology, 40, e36708-e36708 (2018).
19. Azad, T., & Hasin, M. A. A. Capacitated vehicle routing problem using genetic algorithm: a case of cement distribution. International Journal of Logistics Systems and Management, 32(1), p. 132-146 (2019).
20. Ruiz, E., Soto-Mendoza, V., Barbosa, A. E. R., & Reyes, R. Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm. Computers & Industrial Engineering, 133, p. 207-219 (2019).
21. Hosseinabadi, A. A. R., Slowik, A., Sadeghilalimi, M., Farokhzad, M., & Sangaiah, A. K. An ameliorative hybrid algorithm for solving the capacitated vehicle routing problem. IEEE Access, 7, p. 175454-175465 (2019).
22. Zhu, J. Solving Capacitated Vehicle Routing Problem by an Improved Genetic Algorithm with Fuzzy C-Means Clustering. Scientific Programming (2022).
23. Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. Computers & Operations Research, 64, p. 11-27 (2015).
24. Webster, B. L. Solving combinatorial optimization problems using a new algorithm based on gravitational attraction. Ph.D. thesis, Melbourne, Florida Institute of Technology (2004).
25. Lu, H., Shi, J., Fei, Z., Zhou, Q., & Mao, K. Analysis of the similarities and differences of job-based scheduling problems. European Journal of Operational Research, 270(3), p. 809-825 (2018).
26. Rosa, A. F. C. Hybrid Genetic Algorithm based on Fitness Landscape and Principal Component Analysis for the Job Shop Scheduling Problem. 2019, 91f. (Doctoral Thesis – Informatics and Knowledge Management), Nove de Julho University, São Paulo, Brazil (2019).
27. Van Der Maaten, L., Postma, E., & Van den Herik, J. Dimensionality reduction: a comparative. Journal of Machine Learning Research, 10(66-71), 13 (2009).
28. Espadoto, M. Learning Multidimensional Projections with Neural Networks (Doctoral Thesis) University of Groningen, Netherlands (2021).
29. Tayarani-N, M. H., & Prügel-Bennett, A. An analysis of the fitness landscape of travelling salesman problem. Evolutionary computation, 24(2), p. 347-384 (2016).
30. Aguilar-Ruiz, J. S., Giráldez, R., & Riquelme, J. C. Natural encoding for evolutionary supervised learning. IEEE Transactions on Evolutionary Computation, 11(4), p. 466-479 (2007).
31. Poza, J., Moreno, V., Fraga, A., & Álvarez-Rodríguez, J. M. Genetic Algorithms: A Practical Approach to Generate Textual Patterns for Requirements Authoring. Applied Sciences, 11(23), 11378 (2021).
32. Xue, Y., Zhu, H., Liang, J., & Słowik, A. Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification. Knowledge-Based Systems, 227, 107218 (2021).
33. Kumar, A. Encoding schemes in genetic algorithm. International Journal of Advanced Research in IT and Engineering. 2(3), p. 1-7 (2013).
34. Castro, R. E. Optimization of Structures with Multi-Objectives Via Genetic Algorithms. (Doctoral Thesis in Science and Civil Engineering). Federal University of Rio de Janeiro, Brazil (2001).
35. Laporte, G., & Osman, I. H. Routing problems: A bibliography. Annals of operations research, 61(1), p. 227-262 (1995).
36. Christofides, N. Vehicle Routing. In: The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, Lawer, E. L., Lenstra, J. K., Kan, A. H. G. R., Shmoys, D. B. (Eds), John Wiley & Sons (1985).
37. Reinelt, G., Wenger, M. K. Maximally Violated Mod-p Cuts for the Capacitated Vehicle-Routing Problem. Informs Journal on Computing, 18, p. 466-476 (2006).
38. Kalatzantonakis, P., Sifaleras, A., & Samaras, N. Cooperative versus non-cooperative parallel variable neighborhood search strategies: a case study on the capacitated vehicle routing problem. Journal of Global Optimization, 78(2), p. 327-348 (2020).
39. Gillett, B. E., & Miller, L. R. A heuristic algorithm for the vehicle-dispatch problem. Operations research, 22(2), p. 340-349 (1974).
40. Weicker, K. Performance measures for dynamic environments. In International Conference on Parallel Problem Solving from Nature, p. 64-73. Springer, Berlin, Heidelberg (2002).