International Journal of Multicriteria Decision Making

ISSN online: 2040-1078 - ISSN print: 2040-106X
https://www.inderscience.com/ijmcdm

A new matrix form genetic encoding for balanced, compact and connected sectorisation through NSGA-II

Elif Göksu Öztürk, Ana Maria Rodrigues, José Soeiro Ferreira

DOI: 10.1504/IJMCDM.2023.10058115

Article History:

Received:	29 October 2021
Last revised:	18 May 2023
Accepted:	12 June 2023
Published online:	11 October 2023

A new matrix form genetic encoding for balanced, compact and connected sectorisation through NSGA-II

Elif Göksu Öztürk*

INESC TEC – Technology and Science,
Porto, Portugal
and
FEP.UP – Faculty of Economics,
University of Porto,
Porto, Portugal
Email: elif.ozturk@inesctec.pt
*Corresponding author

Ana Maria Rodrigues

INESC TEC – Technology and Science,
Porto, Portugal
and
CEOS.PP,
ISCAP,
Politechnic of Porto,
Porto, Portugal
Email: ana.m.rodrigues@inesctec.pt

José Soeiro Ferreira

INESC TEC – Technology and Science,
Porto, Portugal
and
FEUP – Faculty of Engineering,
University of Porto,
Porto, Portugal
Email: jsf@inesctec.pt

Abstract: Sectorisation refers to dividing a whole into smaller parts, the sectors, to facilitate an activity or achieve some goals. The paper proposes a new matrix form genetic encoding system, called matrix form binary grouping (MFBG), specifically designed for sectorisation and related problems. In MFBG representation, the columns and rows represent sectors and nodes, respectively. As a solution procedure, we followed NSGA-II by contemplating adapted measures for three commonly used criteria (equilibrium, compactness, contiguity) for sectorisation problems. The performance of the MFBG within the NSGA-II is tested from two perspectives: 1) through several experiments on the set of instances;

2) by its comparison with the group-oriented genetic encoding system under the grouping GA. Results showed that the MFBG could find good quality solutions and outperforms the GGA. This confirms that the MFBG is an innovative procedure for dealing with sectorisation problems and an excellent contribution as an alternative encoding technique.

Keywords: sectorisation; matrix form genetic encoding; multi-objective optimisation; NSGA-II; multi-criteria; Pareto frontier; performance metrics.

Reference to this paper should be made as follows: Öztürk, E.G., Rodrigues, A.M. and Ferreira, J.S. (2023) 'A new matrix form genetic encoding for balanced, compact and connected sectorisation through NSGA-II', *Int. J. Multicriteria Decision Making*, Vol. 9, No. 3, pp.250–279.

Biographical notes: Elif Göksu Öztürk has a PhD in Economics from the School of Economics and Management, University of Porto, as well as a BSc and an MA in Economics from the Hacettepe University. Her research interests include development economics, environmental economics, applied economics, geographic information systems, innovation and technological change, operations research and multi-criteria decision making. Between 2019 and 2022, she was a researcher at the Institute for Systems and Computer Engineering, Technology and Science (INESC TEC). Currently, she is an external research collaborator in INESC TEC and a post-doctoral researcher at the University of Minho.

Ana Maria Rodrigues is a Senior Lecturer at the ISCAP – P. Porto (Porto Accounting and Business School-Polytechnic Institute of Porto). She holds a BSc in Applied Mathematics and Computation, a Master's in Applied Quantitative Methods for Management, and a PhD in Engineering and Industrial Management from the Faculty of Engineering, University of Porto. She is a researcher at the INESC TEC – Technology and Science and CEOS.PP (Centre for Social and Organizational Studies). Her research interests are related to sectorisation problems, arc routing, and waste management.

José Soeiro Ferreira is a Professor at the FEUP – Faculty of Engineering, University of Porto (Department of Engineering and Industrial Management) and a research coordinator at the INESC TEC – Technology and Science (Centre for Enterprise Systems Engineering). He holds the Chair Aggregation from FEUP, a PhD in Operational Research from the Technical University of Denmark and a degree in Electrotechnical Engineering from the FEUP. He has been the President of APDIO – The Portuguese Operational Research Society. His main field of activity is operational research/management science. His specific interests cover problem structuring methods and decision and optimisation methods. He authored many articles in international journals and international conferences and coordinated various research and applied projects developed in close cooperation with industrial and service companies.

1 Introduction

Sectorisation refers to dividing a whole – a territory, a network, a large set – into smaller parts, the sectors, considering different criteria for a particular purpose. The application fields of sectorisation problems are diverse. For instance, health and school, political districting, waste collection, maintenance operations, and forest planning are only some of them. Given this diversity, these problems naturally carry multi-criteria characteristics, and their decision-making process involves trade-offs between these criteria.

Various criteria, such as equilibrium, compactness, contiguity, accessibility, capacity, continuity, boundaries, reliability or desirability, are acknowledged in the literature (Martinho et al., 2017; Xie and Ouyang, 2016; Rodrigues and Ferreira, 2015; Bozkaya et al., 2003).

The paper contemplates the most commonly used three criteria for sectorisation problems. These are equilibrium, compactness and contiguity. We carefully evaluated the existing measures for these three criteria, selected the most convenient ones and performed the proper adjustments appropriate for this study.

The solution approach followed was the non-dominated sorting genetic algorithm-II (NSGA-II). Implementing the NSGA-II algorithm for sectorisation is proper since it is common for allocation, location-routing, districting, and sectorisation problems (see Farughi et al., 2020; Zhang et al., 2019; Lara-Caballero et al., 2019; Yin and Suresh, 2018; Vanneschi et al., 2017; Ha et al., 2017; Zou et al., 2016; Prodhon and Prins, 2014). Moreover, it is acknowledged in the literature that NSGA-II gives good results up to three criteria compared to other multi-objective optimisation methods (Zheng et al., 2017).

The central contribution of the work is the proposal of a brand-new two-dimensional genetic encoding system called *matrix form binary grouping* (MFBG) to represent better the solution(s) for sectorisation, allocation, assignment and/or similar problems. In MFBG, the matrix can be considered as an assignment map. The rows represent the nodes, and the columns represent the sectors. Moreover, each sector is designed to have at least one node, and each node must be appointed to no more than one sector. This encoding system is simple, adapted, and expected to promote decision-makers' understanding of the trade-offs between multiple criteria and resolving of sectorisation problems.

The solutions were obtained with the NSGA-II through MFBG genetic encoding system, and several experiments tested the performance in a set of self-generated instances. Given the data scarcity regarding sectorisation problems in the current literature, we decided to create and make the instances available. In total, one hundred instances were generated using both gamma and normal distributions. These instances, with different sizes, are used to test the performance of the proposed MFBG encoding system. The smallest instance created is in the size of 25 nodes, and the largest is in the size of 1,000 nodes. This diversity allowed us to understand the behaviour of our modelling in different sectorisation problems involving three criteria. These instances contain each node's coordinates, quantities, and contiguity maps.

Moreover, we also compared the performance of MFBG and the most commonly used genetic encoding system, called the group-oriented, in the context of sectorisation and related problems, under the same conditions.

This paper is organised as follows: Section 2 briefly introduces sectorisation problems and their multi-criteria nature. Section 3 comprehensively explains the NSGA-II, the selected criteria, and their measures. The section also presents the brand new MFBG genetic encoding system and the main steps of evolutionary algorithms (EAs): selection, crossover, mutation, and the stopping criterion. The instances generated, the results extracted from various experiments and the analyses are exhibited in Section 4. The quality performance of the MFBG with different instance sizes is also displayed in this section. Moreover, the performance comparison of the MFBG and the group-oriented genetic encoding system proposed under grouping GA (GGA) in terms of cardinality, distribution, convergence and spread are presented in Section 5. A discussion and comparison of the CPU times of both genetic encoding systems are also displayed in this section. Finally, Section 6 draws the contributions and conclusions of the work.

2 Sectorisation problems

Sectorisation can be relevant for simplifying and dealing with complex situations. The literature refers to numerous applications that we grouped within three fields: governmental, social, and economic.

Political or communal applications are beneath governmental applications of sectorisation. For example, political districting is an often-implemented application field due to the governmental necessities of sectorisation. Political districting problems aim to divide the territory neutrally and to avoid gerrymandering considering various criteria such as equilibrium, compactness, contiguity and so on (Bozkaya et al., 2003; Chou and Li, 2007; Pukelsheim et al., 2012; Kim, 2018). Other fields studied the most arise due to communal needs, such as the problems with waste collection and water distribution decisions. These applications play a fundamental role in improving the cities' infrastructure and increasing well-being (Mourão et al., 2009). The decision on the divisions of the regions while finding the most efficient path for the waste collection vehicles, and also, satisfying the needs of all service-demanding points constitutes the main purpose of waste collection problems. Moreover, the goal of water distribution problems is, essentially, fair water allocation within the sub-regions with better control over the problematic. These problems can depend on various criteria (Alvisi and Franchini, 2014; Rodrigues and Ferreira, 2015; Yazdandoost and Izadi, 2016; Ríos-Mercado and Bard, 2019).

Sectorisation problems such as energy, microcells, forest planning, sales territories, etc. can be grouped as economic applications. Economic feasibility is one of the most important criteria for finding the most efficient locations for the transmission towers or electricity companies to allocate limited resources like bandwidth (or internet) and electricity (see Lee et al., 2002; Bergey et al., 2003; Brown and Vroblefski, 2004). Moreover, forest planning aims to improve the forest conditions while protecting the wildlife habitat and creating sustainability (see Murray, 1999; Richards and Gunn, 2003; De Assis et al., 2014). The design of sales territories is also beneath economic applications of sectorisation to create compact sectors for the salesperson to travel to clients within the shortest time possible (see Cotta et al., 1995; Jahuira and Cuadros-Vargas, 2003; Lei et al., 2016).

Additionally, applications on police, school, health districting and transportation can be considered as social lines of sectorisation problems. Police districting problems aim to divide the area according to the jurisdiction of the police offices regarding cost, performance and geographical aspects (see D'Amico et al., 2002; Marianov and Fresard, 2005; Camacho-Collados et al., 2015). School districting problems try to find the best divisions of an area considering the balance in the distribution of students among educational institutions according to socioeconomic attributes. Moreover, creating efficiency in transportation costs and considering the schools' capacities are also important criteria for this kind of problem (see Bulka et al., 2007; Bouzarth et al., 2018). Likewise, health districting aims to provide the fastest transportation to the patients keeping the balance in the workloads of the hospitals and facing the patients' needs (see Koduru et al., 2004; Farughi et al., 2019; Yanık et al., 2019). Finally, safe and efficient air traffic and air services can also be possible using sectorisation methods. Essentially, airspace sectorisation is one of the major application fields given its international and intranational importance in transportation (see Degtyarev et al., 2009; Tang et al., 2012).

As is seen, sectorisation problems are diverse and usually complex due to the contemplation of several criteria. The literature acknowledged different methods, over time, to deal with the difficulty of their resolution (Nagy and Salhi, 2007; Jourdan et al., 2009). Heuristic and metaheuristic methods are often used due to their speed and applicability whatsoever the size of the problems. They aim to find a relatively better solution among all feasible solutions, although they may not find the best solution (Pirlot, 1996; Jourdan et al., 2009; Salhi, 2017).

Problems with multiple criteria can be handled by a single-objective or a multi-objective approach. The former refers to collecting all the criteria within a single composite objective function by considering some weights according to the relative importance of each criterion. On the other hand, the latter provides a set of feasible solutions by evaluating the multiple criteria simultaneously by locating them on Pareto-frontiers.

Multi-objective optimisation gives decision-makers the chance to choose from multiple trade-off solutions. Deb (2001, p.22) acknowledged that "it is intuitive to realise that single-objective optimisation is a degenerate case of multi-objective optimisation since multi-objective optimisation involves multiple objectives." Thus, multi-objective optimisation is a superior procedure when it is compared to single-objective optimisation.

The generalised version of the multi-objective optimisation problem can be defined in equation (1):

$$\text{Min } \mu = f(x) = (f_1(x), f_2(x), \ldots, f_k(x)) \text{ subject to } x \in X \tag{1}$$

Here μ is the objective vector, and f is the mapping function from the solution space X to an objective space R. Moreover, $f_i(x)$ is the i^{th} objective and x is a vector of decision variables.

Let us assume a minimisation problem (all objectives are for minimisation). For instance, if there are only two feasible solutions, x and y, the solution x dominates the solution y if and only if $f_i(x) \leq f_i(y)$ for $i = 1, 2, ..., k$ and $f_j(x) < f_j(y)$ for at least one objective function j. Hence, x shows the Pareto optimal solution. In other words, to be a Pareto optimal solution, a solution should strongly dominate at least one other solution while not being dominated by any other solutions in the solution space for each objective. Thus, a solution can only be Pareto optimal if it performs better than other solution(s) in the solution space for the direction of the objective (min. or max. problem). The set of all non-dominated solutions is called a Pareto optimal set. The values of the objective functions which belong to the Pareto optimal set constitute the Pareto front (Konak et al., 2006).

3 NSGA-II using MFBG genetic encoding

There is a variety among multi-objective optimisation methods within the scope of EAs. The idea of implementing multi-objective optimisation on EA was brought by Schaffer (1985) while proposing the vector evaluated GA. However, a breakthrough multi-objective method was created by Goldberg et al. (1989), called a multi-objective evolutionary algorithm (MOEA), which uses the concept of domination. Several researchers followed MOEA for different applications. Non-dominated sorting genetic algorithm (NSGA), proposed by Srinivas and Deb (1994), is one of the modified algorithms of MOEA. The NSGA has been criticised over time for having high computational complexity, not having an elitism factor, and needing a user-defined sharing parameter. NSGA-II was developed as a more advanced version of NSGA by Deb et al. (2002). In this version, all the criticisms have been erased (Deb, 2001). Following these improvements, NSGA-II became one of the most popular multi-objective GA methods (Song and Chen, 2018). This is one of the reasons why we also follow NSGA-II to solve sectorisation problems besides those indicated in Section 1.

NSGA-II helps to create Pareto fronts iteratively through new populations as non-dominated sorting-base algorithm (Deb et al., 2002). The frontiers are composed according to the two necessary parameters. The first parameter is the domination count n, which counts how many other solutions dominate a solution. For instance, if two other solutions dominate solution p, n_p is equal to two. The second parameter is the set of dominating members S. This set includes the solutions dominated by a solution. For example, if solution z dominates the solutions p and t, S_z is composed of $\{p, t\}$. The solutions with the lower count of domination n constitute the first Pareto front. In other words, solutions not dominated by other solutions create the first Pareto front. Solutions that are only dominated by the solutions in the first front constitute the second. Likewise, solutions only dominated by the first and second fronts establish the third front. The procedure continues until all the solutions are in a front set (Deb et al., 2002). Thus, the pool of superior solutions is gauged using these two parameters in the NSGA-II algorithm.

Moreover, the main concepts, namely, the genetic encoding system, selection, crossover, mutation and the stopping criterion should be well-considered due to their effect on the quality and the performance of the final solutions. During the remainder of this section, we explain the methods followed in detail and provide a comprehensive definition of the three criteria (i.e., equilibrium, compactness and contiguity) used.

3.1 'MFBG' genetic encoding

In GA, each solution is called a chromosome and has to be encoded in a certain format proper for the problem since the chromosome format directly affects the results (Tsai et al., 2015). Several types of genetic encoding have been proposed and implemented. For example, some are binary strings, real-valued vectors, variable-length, permutation, group-oriented, finite state or parse-tree (Agustín-Blas et al., 2009; Tsai et al., 2015). Among these encoding systems, binary string and group-oriented, are relevant to our approach.

Binary string encoding, presented by Holland (1975) for the first time, is one of the most commonly used chromosome types in GA applications. In this encoding system, chromosome genes can only take 0 or 1. Moreover, a group-oriented genetic encoding system is first presented by Falkenauer (1992) under the GGA. This type of encoding system separates each chromosome into two parts. While the first part represents the assignment, the second part of the chromosome represents the group part. The number of groups is flexible and can be different for each chromosome. On the other hand, this number can also be fixed in advance as it is equal for each chromosome. In the assignment part, each gene is attributed to one and only one of the groups in the group part. For example, if 11 elements are to be assigned to 3 classes, then, according to the grouping genetic encoding system, the shape of the chromosome is as follows [1 2 3 3 2 1 1 2 2 1 1 | 1 2 3]. Class one consists of elements 1, 6, 7, 10, 11, class two consists of 2, 5, 8, 9, and the last class comprises elements 3 and 4. Empirical applications of the GGA using this form of encoding also appear in the literature often (Brown and Vroblefski, 2004; Agustín-Blas et al., 2009, 2012; Zou et al., 2016).

Although the array format genetic encoding system is typical, it is also possible to see different formats on the solution representations. Fox and McMahon (1991) first represented the matrix form chromosome to the best of our knowledge. The authors used a square form matrix chromosome to solve specifically travelling salesperson problems. In that matter, they employed a two-dimensional binary form chromosome to represent the sequence of the nodes to travel. This method is chosen by other authors to be used on the same problem (Osorio et al., 2002). Moreover, Wallet et al. (1996) also proposed a matrix form chromosome to simplify the solutions for problems with two-dimensional nature. Likewise, Tsai et al. (2015) presented a new matrix form encoding scheme proper for two-dimensional airspace scheduling problems. Their results indicate that the proposed encoding is effective on solutions.

Albeit it is recognised that using matrix form chromosomes can improve one's understanding of the solutions (Wallet et al., 1996), there is still a significant gap in the literature on the implications of this formatting (Chen et al., 2016).

In this work, we put effort into closing this gap and adopt the idea of a matrix form chromosome to develop a new encoding scheme proper for sectorisation, assignment, allocation or similar problems. Doing this aims to represent the solutions more clearly and simplified. Therefore, the paper proposes a new genetic encoding system, the 'MFBG', combining group-oriented and binary-string forms within a two-dimensional framework. A binary string in each row of the matrix encodes the problem. Moreover, a row's value (i.e., 0 and 1) represents a grouping state.

An illustrative example of the proposed matrix form chromosome is presented in equation (2). The rows represent the nodes and the columns are the sectors in a sectorisation problem. For instance, node 1 is assigned to sector 1, while node 2 is assigned to sector 3.

$$MFBG = \begin{bmatrix} 1\,0\,0 \\ 0\,0\,1 \\ 0\,1\,0 \\ 0\,1\,0 \\ 1\,0\,0 \\ 1\,0\,0 \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix} \tag{2}$$

Assume that x_{ij} represents the assignment state of the node i into sectors j.

$$x_{ij} = \begin{cases} 1 & \text{if node } i \text{ is assigned sector } j \\ 0 & \text{otherwise} \end{cases}$$

According to this representation, x_{ij} equals 1 if the node i is assigned to the sector j, and 0 otherwise. There are two strict restrictions in this encoding system:

1 Each node can be assigned to only one sector.

$$\sum_j x_{ij} = 1, \ \forall i$$

2 At least one node has to be assigned to each sector.

$$\sum_i x_{ij} \geq 1, \ \forall j$$

A solution is feasible if and only if it verifies both conditions simultaneously.

We expect that the MFBG brings clear advantages for the use of GA in small, medium or large-scale sectorisation problems given its design.

3.2 Criteria and definitions

As we established in Section 2, sectorisation problems naturally involve some criteria that must be considered to attain admirable decisions.

This paper focuses on three criteria, equilibrium, compactness and contiguity, which are the most commonly used in sectorisation problems.

3.2.1 Equilibrium

Equilibrium means creating a similarity between sectors during division decisions. For instance, a balance in workload for the salesperson, a similar number of voters for the political districts, equivalence on the number of students in the schools or proportional partition of patients in hospitals can only be provided by an equilibrium factor. The procedure for equilibrium by Rodrigues and Ferreira (2015) is utilised, with a slight difference, which can be observed below:

Assume that K is the total number of sectors, and each sector has a quantity q_j, where $j = 1, 2, ..., K$. Equation (3) shows the mean value for the quantity of each sector.

$$\bar{q} = \frac{\sum_{j=1}^{K} q_j}{K} \tag{3}$$

where

$$q_j = \sum_i x_{ij},\ \forall j$$

Equation (4) shows the standard deviation controlled by the sector number. As is seen, the equilibrium is a proxy for the deviations of the quantities from the mean quantity of all sectors. The smaller the equilibrium score, the better the model is. In other words, we consider a minimisation problem for this criterion.

$$std'_{eq} = \sqrt{\frac{1}{K-1}\sum_{j=1}^{K}(q_j - \bar{q})^2} \tag{4}$$

Equilibrium may not only be related to the number of nodes assigned to the sectors. The demand may also affect the equilibrium factor in some sectorisation problems, such as waste collection or travelling salesperson problems. Overseeing the quantity of demand at a point can be misleading. As it is represented in equation (5), each solution column is multiplied by the quantity (e.g., demand or service requirement) matrix to avoid this. For instance, assume that each node assigned to a sector in the $MFBG$ encoding system, represented in equation (5), has a different quantity of demand. The quantity matrix for these points is represented as Y (i.e., $Y = [y_{1i}]_{i=1,..,N}$) in equation (5). As is seen in the quantity matrix, point one, which is assigned to sector one, has two as demand. Likewise, point two, assigned to sector three, has four as demand. The same interpretation can be made for each point.

$$Y \times MFBG = \begin{bmatrix} 2\,4\,1\,2\,3\,1\,1\,4 \end{bmatrix} \times \begin{bmatrix} 1\,0\,0 \\ 0\,0\,1 \\ 0\,1\,0 \\ 0\,1\,0 \\ 1\,0\,0 \\ 1\,0\,0 \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix} \tag{5}$$

Although three nodes compose sector 1, when we consider the quantity matrix, the total demand is 6. Thus, the numerator of equation (3) must be rewritten as below.

$$q_j = \sum_{i=1}^{N} y_{1i} \times x_{ij},\ \forall j$$

This way, we avoid the underestimation of equilibrium criteria. Consequently, the proposed MFBG encoding applies to different kind of sectorisation problems.

3.2.2 Contiguity

Contiguity means existing linkages between the nodes within the same sector. We follow Rodrigues and Ferreira (2015) and use a matrix-based method for this criterion to calculate contiguity. This is a symmetric matrix with the size of the total nodes ($N \times N$). The contiguity matrix is represented as follows:

$$M = \begin{pmatrix} 0 & m_{12} & \cdots & m_{1N} \\ m_{21} & 0 & \cdots & m_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N1} & m_{N2} & \cdots & 0 \end{pmatrix}$$

This matrix is binary, with

$$m_{wi} = \begin{cases} 1 & \text{if there is a linkage between point } w \text{ and point } i \\ 0 & \text{otherwise} \end{cases}$$

This representation shows the overall contiguity frame between all nodes in the space (or territory). However, this can be misleading if this matrix is not sector-specific. For instance, suppose that w, i and f are nodes assigned to the same sector (j). If nodes w and i are linked, and nodes w and f, nodes i and f should also be considered connected due to their path through node w.

To follow such an approach, we consider a sector-based sub-matrix with the size $(n_j \times n_j)$ where n_j is the number of nodes assigned to sector j. The sub-matrix is:

$$M^j = \begin{pmatrix} 0 & m^j_{12} & \cdots & m^j_{1n_j} \\ m^j_{21} & 0 & \cdots & m^j_{2n_j} \\ \vdots & \vdots & \ddots & \vdots \\ m^j_{n_j1} & m^j_{n_j2} & \cdots & 0 \end{pmatrix}$$

This matrix is built according to all existing paths, as explained above, with the entries defined as:

$$m^j_{wi} = \begin{cases} 1 & \text{if there is a path between point } w \text{ and point } i \text{ in sector } j \\ 0 & \text{otherwise} \end{cases}$$

Thus, given that all points linked to each other with a direct or an indirect path are connected, we should consider the number of connected components (R) to calculate contiguity for this sub-matrix.

The expression to evaluate the contiguity in light of all these explanations can be observed in equation (6). Here the numerator shows the sum of all paths between the nodes for the number of connected components (n_r, $r = 1, ..., R$). Moreover, the denominator shows the maximum number of undirected linkages in a sector according to the assigned points. That means c_j equals 1 if all the points are connected (or there is only one connected component in the sector).

$$c_j = \frac{\sum_{r=1}^{R} (n_r \times (n_r - 1))}{n_j \times (n_j - 1)} \tag{6}$$

Contiguity is then represented considering the size of the sectors (i.e., weights of the sectors) for the number of points assigned to the same sector. That is expressed in equation (7).

$$\bar{c} = \frac{\sum_{j=1}^{K} c_j . n_j}{N} \tag{7}$$

This criterion occurs on a scale between 0 and 1. Moreover, scores close to 1 show a better status for contiguity. This work deals with contiguity by minimising $(1 - \bar{c})$.

3.2.3 Compactness

Compactness is difficult to measure since it is possible to find several definitions for different research purposes. In the current work, we consider compactness as the density of the sectors. The method followed to measure compactness can be observed in equation (8).[1]

$$d = \sum_{j=1}^{K} dist(o_j, p_j) \tag{8}$$

In equation (8), the d shows the sum of the distance between the centroid of the sector o_j and the furthest point to the centroid p_j for j sectors, where $j = 1, 2, ..., K$. Smaller values obtained for compactness represent better results. In other words, we seek to minimise the maximum distance within sectors.

3.3 Solution procedure

In NSGA-II, the process starts with the initialisation of the population. After calculating three criteria for each solution in the initial population, Pareto frontiers are composed. Afterwards, each generation consists of the following steps: selection, crossover, mutation, elitism and stopping criterion.

Algorithm 1 shows the pseudocode of NSGA-II.

Algorithm 1 Pseudocode of NSGA-II

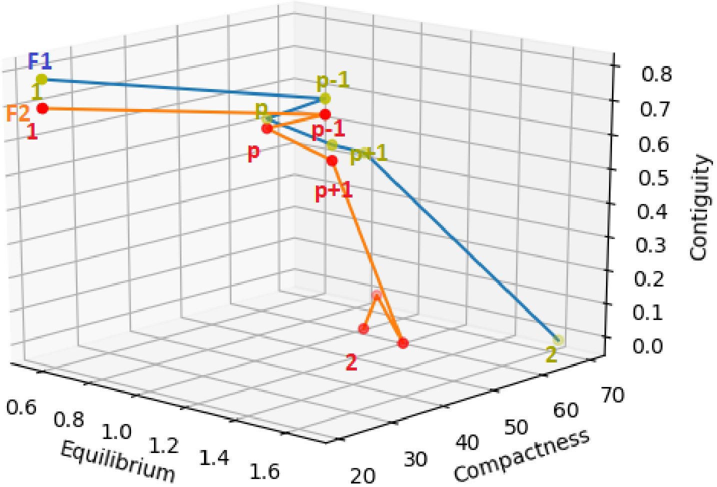
```
Generate $N$ feasible solutions and insert them into population ($Pop_{size} = N$)
$f_{equity}(x)$, $f_{contiguity}(x)$, $f_{compactness}(x)$
Non-dominated sorting ($t = 0$) min. (equity, contiguity, compactness)
Calculate crowding distance of each frontier
while $\underline{not}\ terminate$ do
    while $Pop_{size} \neq N \times 2$ do
        Select parents through tournament selection
        Create two off-springs using a two-point crossover in each turn
        Mutate off-springs (for selected $P_{mut}$)
        Merge off-springs into population
        $Pop_{size} := Pop_{size} + 2$
    end while
    for each individual $x$ in the current population do
        $f_{equity}(x)$, $f_{contiguity}(x)$, $f_{compactness}(x)$
        Non-dominated sorting ($t = t+1$)
        Calculate crowding distance of each frontier
    end for
    Generate set of non-dominated vector along Pareto fronts
    while $Pop_{size} \neq N$ do
        Delete the solution according to frontier or the crowding distance
        $Pop_{size} := Pop_{size} - 1$
    end while
    for each individual $x$ in the population do
        $f_{equity}(x)$, $f_{contiguity}(x)$, $f_{compactness}(x)$
        Non-dominated sorting
```

```
        Calculate crowding distance of each frontier
    end for
    Output the results
end while
```

3.3.1 Selection

Selection occurs after the solutions are placed on the Pareto frontiers according to their performance on the criteria separately. The tournament selection is followed, which means, between two individuals picked at random, the superior individual is chosen according to the location on the Pareto frontiers (i.e., according to their fitness). If a solution is in a better Pareto frontier, that solution is superior. If the selected solutions are in the same frontier, the crowding distance (CD) concept of NSGA-II is called for to pick the superior solution.

Figure 1 Visual representation of Pareto front sets and CD (see online version for colours)



CD shows the concentration of the solutions. CD should be calculated for each solution in each frontier. If CD occurs small, it covers a small number of solutions (i.e., less diversity among solutions). On the other hand, higher CD values are more representative of the population (Deb et al., 2002). Equation (9) describes how to calculate CD for the three elected criteria. Figure 1 illustrates the CD process again for the three selected criteria. The example has a solution with two Pareto frontiers, F1 representing the first frontier, F2 the second frontier and points 1 and 2 representing the endpoints, set to infinity. The CD value of all the solutions in the Pareto frontiers must be measured. Solutions that appear on the same frontier are evaluated on their own merits. Since each frontier includes three bands of values for the three criteria, the total CD is the sum of the CD values of each criterion [equation (9)].

$$CD^{p}_{eq} = \frac{|f^{(p+1)}_{eq} - f^{(p-1)}_{eq}|}{f^{(\max)}_{eq} - f^{(\min)}_{eq}},$$

$$
\begin{aligned}
CD^{p}_{com} &= \frac{|f^{(p+1)}_{com} - f^{(p-1)}_{com}|}{f^{(\max)}_{com} - f^{(\min)}_{com}} \\
CD^{p}_{con} &= \frac{|f^{(p+1)}_{con} - f^{(p-1)}_{con}|}{f^{(\max)}_{con} - f^{(\min)}_{con}} \\
CD^{p} &= CD^{p}_{eq} + CD^{p}_{com} + CD^{p}_{con}
\end{aligned} \tag{9}
$$

Thus, in the selection process, if the two solutions are in the same Pareto frontier, according to these explanations, the one with a higher CD value is selected as superior.

3.3.2 Crossover

Crossover [i.e., mating the solutions to create new solutions (off-spring)] is a GA-specific operator which cannot be seen in any other heuristic method (Chen et al., 2016). This step comprises of two stages:

1 selection of parents

2 mating.

After the selection of parents, the mating process starts. There are different crossover methods, such as single-point, random block, multi-point, randomised, uniform or substring crossover (Wallet et al., 1996; Tsai et al., 2015; Chen et al., 2016), to mate the parents. The multi-point crossover operator is used. This operator is uncommon in two-dimensional GA applications (i.e., matrix form genetic encoding).

It is possible to observe the multi-point crossover method in Figure 2. As is seen, we randomly select points and switch the selected rows between matrices (parents) to create two brand new off-springs (i.e., new solutions).

Figure 2 Visual representation of crossover process using MFBG

$$
\begin{bmatrix} 1\,0\,0 \\ 0\,0\,1 \\ \mathbf{0\,1\,0} \\ \mathbf{0\,1\,0} \\ 1\,0\,0 \\ \mathbf{1\,0\,0} \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix}
\Longleftrightarrow
\begin{bmatrix} 0\,0\,1 \\ 1\,0\,0 \\ \mathbf{0\,1\,0} \\ \mathbf{0\,0\,1} \\ 1\,0\,0 \\ \mathbf{0\,1\,0} \\ 0\,1\,0 \\ 1\,0\,0 \end{bmatrix}
\quad\longrightarrow\quad
\begin{bmatrix} 1\,0\,0 \\ 0\,0\,1 \\ 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,1 \\ 0\,0\,1 \end{bmatrix}
\qquad
\begin{bmatrix} 0\,0\,1 \\ 1\,0\,0 \\ 0\,1\,0 \\ 0\,1\,0 \\ 1\,0\,0 \\ 1\,0\,0 \\ 0\,1\,0 \\ 1\,0\,0 \end{bmatrix}
$$

Moreover, the MFBG genetic encoding system assures that a single basic unit can only be assigned to one sector during the crossover. In addition, it is simple to determine if a sector is vacant by summing the columns after the crossover. Such a benefit of the MFBG makes it an advantageous alternative compared to other methods.

3.3.3 Mutation

The mutation can be convenient to increase the diversity in the pool of solutions. This operator is used to increase the diversity in the population. The mutation occurs according to the mutation probability imposed on the population. Since high probability could randomise the search, the small mutation probability is considered in classical GA (Liu and Kroll, 2016). Mutation probability shows the potential alterations in chromosomes (i.e., solutions). We implement the mutation on only off-springs with a very small probability.

More specifically, the purpose of the mutation is to find out whether there is a better assignment for a customer to achieve a more convenient sectorisation eventually. As mentioned in Subsection 3.1, each matrix shows a solution, and each row of the matrix represents the assignment status of a node. The mutation may change this assignment status according to mutation probability. Thus, the algorithm takes a solution and decides, row by row, whether there will be a change in the assignment status of a node. Therefore, the mutation may change a solution totally, or only one row of a solution or even may not change anything. This occurrence totally depends on the mutation probability. The mutation probability applied is 0.05.

3.3.4 Stopping criterion

A stopping criterion is very convenient for computational efficiency since it helps an algorithm to stop when there are no more improvements in the obtained solutions. In multi-objective optimisation, results advancement between iterations should be evaluated (Martí et al., 2016).

The stability measure is based on the density of the non-dominated solutions presented by Rudenko and Schoenauer (2004). The authors have created an NSGA-II specific stopping criterion to gauge the standard deviation of maximal CD among the non-dominated individuals for a selected number of generations. The algorithm stops if the standard deviation is smaller than the selected threshold. Equation (10) denotes this process.

$$\sigma_L \equiv \sqrt{\frac{1}{L}\sum_{j=1}^{L}(d_l - \bar{d_L})^2} < \delta_{lim} \tag{10}$$

Here, d_l is the maximal CD value in the l^{th} iteration among the non-dominated solutions. Moreover, $\bar{d_L}$ is the mean maximal value over L iterations. Two important parameters should be selected in advance. These are L, which represents the length of the time window, and the threshold value δ_{lim}.

Rudenko and Schoenauer (2004) stated that these parameters should be selected according to the chosen population number. When the population increases, L should also be increased as well as δ_{lim} should be decreased.

The threshold value is δ_{lim} = 0.04 and the time window L = 20 for the population size of 50.

4 Data and results

In this section, we present the data, the scope of the experiments, and the results in detail. The first subsection extensively explains the approach to generating the data. The second subsection confers the results and provides an overall view of their performance and some examples of how the solutions look when using the MFBG genetic encoding system. The algorithm is implemented through Python 3.7, and the method is executed on a PC with an Intel Core i7-8550U at 1.8 GHz and Win X64 operating system. The results presented in this section focus on representing the quality of the obtained solutions.

4.1 Data

Given data scarcity, we tested the MFBG genetic encoding system on artificially created data to understand whether it works well. We generated one hundred different instances. Since the purpose is to design an algorithm appropriate for small and large sectorisation problems, the instances were conveniently designed. The smallest sample has 25 nodes, and the largest has 1,000 nodes. Coordinates, quantities and contiguity maps are all generated for each instance.

To improve the experiments' robustness, we used both gamma and normal distributions to generate the instances. The shape and scale parameters necessary for gamma distribution were created randomly for every single node in every instance. Likewise, the mean and the variance parameters necessary for the normal distribution were also produced randomly. We selected two values from two sets:

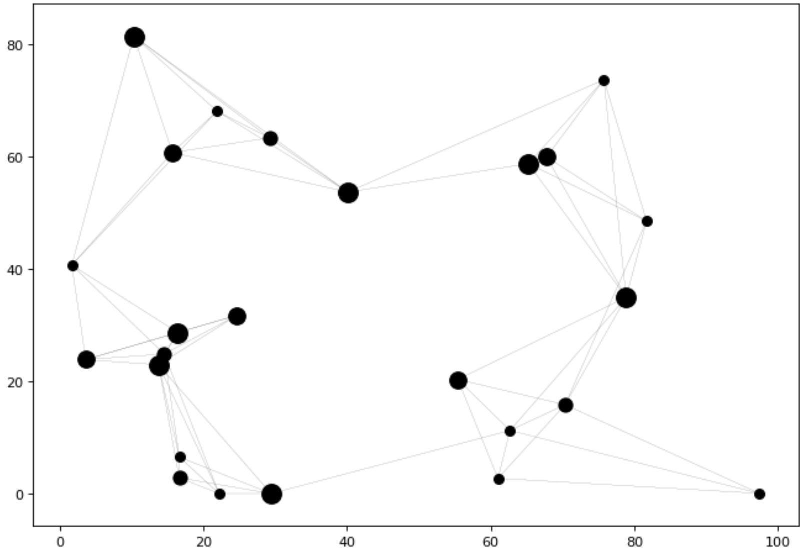
1 between 1 and 10 for parameters
2 between 25 and 100 for observations.

Different mean and variance parameters were generated as much as the first selected number. Then, as much as the second number for each parameter set, sets of nodes were produced. Finally, they collected them within the same instance. For example, if the first selected number is 4, and the second number is 20, then the instance size is 80 nodes, consisting of four clusters of 20 nodes produced by different mean and variance parameters. Thus, a group of nodes with different mean and variance produced each normally distributed instance. The aim was to generate instances similar to the real case data.

The quantities were obtained using a uniform distribution. The contiguity maps were created considering the connected graph theory and the distances between nodes. We selected a number between 3 and 10 and linked each node to the closest nodes to the maximum number of connections defined. After the map generation, according to this rule, we calculated the connectivity components of the whole space. If more than one component was detected, we added a link between the two closest nodes of separate components to guarantee a connected graph. The contiguity factor was evaluated for every sector (or column) of each chromosome for this map. In this way, connectivity was guaranteed in all sectors. Figure 3 depicts an example.

We placed a special effort to create unbiased instances due to the relevance of testing the algorithm on neutral data and getting a good evaluation of the quality of the results. These instances are available at: https://drive.inesctec.pt/s/NS47qnZEmYPwEQP.

Figure 3 Connected graph



4.2 Results

We performed experiments for a different number of sectors to test the MFBG genetic encoding system using NSGA-II. The size of the population and the number of generations are kept the same for all tests. We first separated the instances as small, medium and large. Instances up to 100 nodes are considered small, between 100 and 500 are medium, and those with more than 500 nodes are considered large. The predetermined sectors (S) used to test the instances are shown below.

$$S_{small} = \{5,\ 10\}$$
$$S_{medium} = \{5,\ 10,\ 15,\ 20\}$$
$$S_{large} = \{10,\ 20,\ 30,\ 50\}$$

In GA, as it happens in nature, the population tends to converge to a plateau over time. In other words, individuals in the population start to become similar over generations. The mutation increases the diversity in the population. We used P_{mut} = 0.05 as the mutation rate. The multi-point crossover we implemented, is quite powerful and efficient for the MFBG genetic encoding system. Thus, using a higher rate of mutation probability was not necessary. Moreover, the execution of several experiments on higher mutation rates showed us that the approximation of the solutions, requisite for the stopping criterion, occurs delayed. Thus, it may have decreased the time efficiency of the model ultimately.

We run the model with a population size of 50. The overall complexity of the NSGA-II is $O(MN^2)$, where M is the number of criteria and N is the population size (Deb et al., 2002). The population size was kept at 50, since the previous efforts and existing literature (see Sarrazin and Smet, 2016) showed that it was good enough to obtain reliable results for a multi-criteria model. Likewise, we selected the parameters of the stopping criterion according to the population size, as they should be compatible with each other to avoid immature or delayed stopping of the algorithm (Rudenko and Schoenauer, 2004). Thus, δ_{lim} = 0.04 and L = 20 are the values for the two necessary parameters of the stopping criterion.

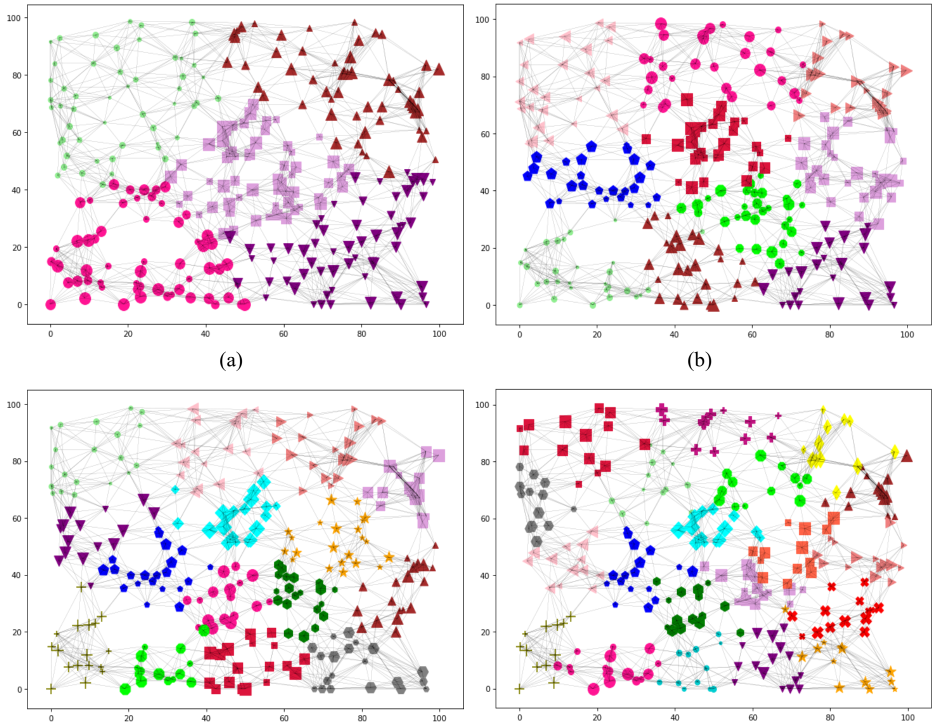
Table 1 provides an overall look at the results. It provides the possibility to assess the behaviour of NSGA-II on all instances for each predefined sector, using the MFBG genetic encoding. In addition to the best and worst results for the instance groups (small, medium, large), the average of all results as a proxy for the quality of the solutions can be observed in that table. Moreover, it is possible to access all the results for each instance in the link: https://drive.inesctec.pt/s/jH5qjAjYFSqc7cn.

Table 1 Overall evaluation of results tested

Instance	*Sector*	*Indicator*	*Equilibrium*	*Compactness*	*Contiguity*
Small		Average	1.764	127.514	0.039
	5	Best	0.0	7.693	0.0
		Worst	5.029	188.534	0.377
Small		Average	2.463	165.229	0.055
	10	Best	0.601	4.296	0.0
		Worst	5.342	302.556	0.216
Medium		Average	2.079	175.747	0.009
	5	Best	0.5	81.088	0.0
		Worst	13.893	393.056	0.123
Medium		Average	2.393	247.289	0.027
	10	Best	0.333	99.347	0.0
		Worst	13.668	564.769	0.151
Medium		Average	3.486	294.506	0.0314
	15	Best	0.9645	103.318	0.0
		Worst	10.876	874.719	0.388
Medium		Average	3.233	290.958	0.017
	20	Best	1.051	111.934	0.0
		Worst	6.249	635.552	0.187
Large		Average	2.714	295.269	0.031
	10	Best	0.0	175.272	0.0
		Worst	7.91	479.644	0.249
Large		Average	3.777	391.391	0.022
	20	Best	0.917	289.775	0.0
		Worst	12.559	767.266	0.108
Large		Average	4.857	591.553	0.054
	30	Best	1.433	344.722	0.0
		Worst	13.703	1,765.095	0.344
Large		Average	4.182	571.411	0.0261
	50	Best	1.998	364.063	0.0
		Worst	6.375	1,155.572	0.195
L = 20		δ_{lim} = 0.04		Pop_{size}= 50	P_{mut} = 0.05

As is seen in Table 1, we obtained qualified solutions. The average results for each criterion are persuasive. For instance, equilibrium values are between 2 and 4, while contiguity values are very close to 0, which is the best case. Finally, compactness values show an upward tendency for bigger sector sizes overall. When considering the compactness measurement, this tendency is normal. The results in Table 1 can be a comparative resource for future studies employing our instances and the chosen parameters.

Figure 4 Sectorisation using the MFBG genetic encoding on a medium instance, (a) 285 nodes – 5 sectors (b) 285 nodes – 10 sectors (c) 285 nodes – 15 sectors (d) 285 nodes – 20 sectors (see online version for colours)



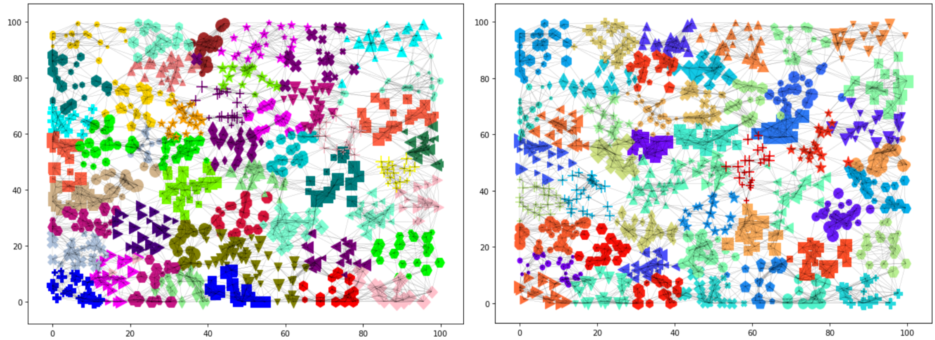
(a) (b)

(c) (d)

Node	Pop_{size}	P_{mut}	*Gen.*	*Sector*	*Equation*	*Comp.*	*Cont.*	*Time*
285	50	0.5	80	5	0.05	154.168	0.0	00:02:13
			180	10	0.33	225.069	0.007	00:03:59
			240	15	3.28	251.069	0.0	00:06:16
			400	20	1.61	297.708	0.007	00:13:23

Figure 4 shows an example from the solution set of a middle-size instance (285 nodes) for different sector sizes (i.e., 5, 10, 15 and 20). The numerical results of the visually represented solutions are also presented in Figure 4. The algorithm can almost find the best possible solution for contiguity (i.e., 0.0) for all sector sizes tested. Furthermore, in the performance of compactness, an expansion in the final results is normal when a larger number of sectors is in question, given the measure used for this criterion [see equation (8)]. Equilibrium values obtained are also satisfying for all cases in this example. The generation in which the algorithm stops and the related computation times are displayed in this table's fourth and last columns. The algorithm stops when no further significant improvements in the final solutions are attained.

Figure 5 Sectorisation using the MFBG genetic encoding on a large instance, (a) 1,000 nodes – 30 sectors (b) 1,000 nodes – 50 sectors (see online version for colours)



(a) (b)

Node	Pop_{size}	P_{mut}	*Gen.*	*Sector*	*Equation*	*Comp.*	*Cont.*	*Time*
1,000	50	0.05	120	30	2.03	347.498	0.0	00:29:20
			260	50	6.67	463.357	0.002	01:03:29

It is also important to show whether the MFBG encoding system works well for large instances. Thus, we also put effort into obtaining robust, fast and good results on big cases. Figure 5 shows an example of the algorithm behaviour for an instance of 1,000 nodes. Figure 5(a) shows the division of the territory within 30 sectors and Figure 5(b) within 50 sectors. The results for equilibrium are better for 30 sectors division than 50 sectors division in this example. Furthermore, the sectors are almost entirely connected. Compactness, given the measurement, depends on the sector size.

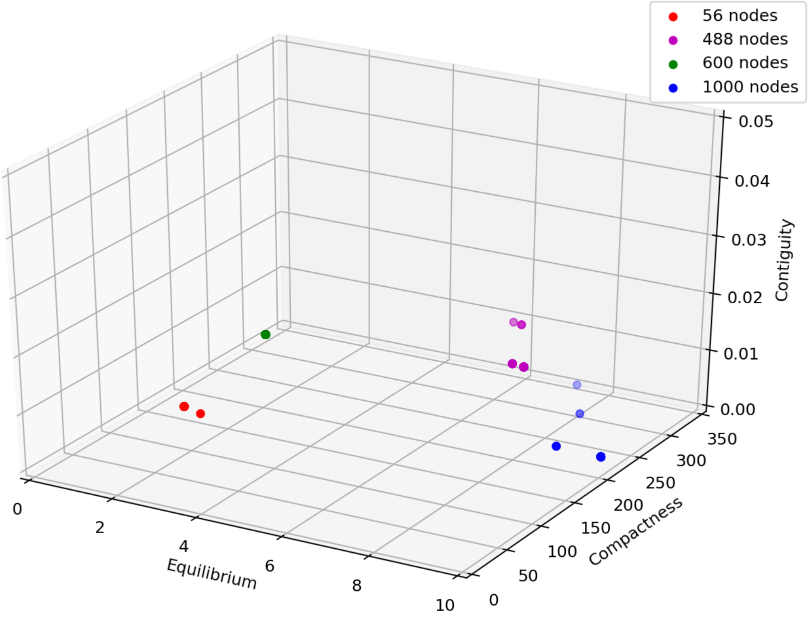
Besides these specific cases presented in Figures 4 and 5, the general behaviour of the solutions in four example instances can be observed in Figure 6. Using the MFBG genetic encoding system with three criteria reaches an overall approximation of a plateau solution. There is no domination among the solutions since all the solutions occur either very similar or identical.

In Figure 6, although there are 50 final solutions in the Pareto front, for the 600 nodes instance, all the solutions are collected in a single point. Besides, the instance with 56 nodes shows only two different solutions, and the instances with 1,000 and 488 nodes ended up with only four different solutions. Moreover, these solutions are still very similar to each other. This situation is valid in all instances.

The results show that it is challenging to find feasible solutions while satisfying the three criteria simultaneously. Although the model is flexible to alterations in size and parameters, results are highly dependent on the data used.

Regarding all the experiments in this section, we can conclude that it is possible to find good quality solutions using the MFBG genetic encoding system.

Figure 6 The convergence of the results (see online version for colours)



5 Performance comparison and discussion

In this section, we compared the performance of the MFBG and group-oriented genetic encoding systems. Given that the group-oriented genetic encoding system is explicitly proposed for the GGA by Falkenauer (1992), we used the GGA as the abbreviation of the group-oriented encoding system during this section.

We implemented both the GGA and the MFBG encoding system in NSGA-II to compare their performances in six different performance metrics and the CPU times. We used the same parameters and same decoding methods (i.e., selection, crossover and mutation) to compare the performance of the two genetic encoding systems.

The performance metrics evaluate the performance of Pareto optimal solutions according to three goals. These are

1 minimising the distance between the actual Pareto front (or reference Pareto front) and the Pareto optimal set of solutions (or approximation Pareto front)

2 obtaining a good distribution of the solutions in the objective space

3 maximising the extent of the Pareto optimal solutions (Zitzler et al., 2000).

Several performance metrics exist to reach the goals mentioned above. It is possible to group these metrics within six categories:

1 cardinality

2 convergence

3 distribution

4 spread

5 convergence and distribution

6 spread and distribution (Yen and He, 2013; Audet et al., 2020).

We evaluated the performance of the two genetic encoding systems regarding all six categories.

To compare the performance of the MFBG over the GGA, we selected ten instances of different sizes and ran each instance seven times using the same parameters for both. It is possible to see the parameters in Table 2.

Table 2 The parameters and instances used for performance comparison

Instance	*Node*	Pop_{size}	P_{mut}	*Gen.*	*Sector*
Gamma2	690	50	0.05	500	30
Gamma3	56				5
Gamma8	873				30
Gamma9	432				10
Gamma10	102				10
Gamma11	288				10
Gamma14	204				10
Gamma18	528				30
Gamma28	350				10
Gamma49	1,000				30

As is seen in Table 2, we used 500 generations as the stopping criterion. Additionally, during these experiments, we sectorised small instances within five sectors, medium-size instances within ten sectors, and big-sized instances within 30 sectors. The mutation rate and population size are kept at 0.05 and 50, respectively.

It is important to mention that we created a reference Pareto front by considering all Pareto fronts obtained for two encoding systems and seven trial runs. Actual or reference Pareto front is essential given that the expressions of almost all the performance metrics require that information.

Furthermore, we adopted all the performance metrics from Yen and He (2013). To know the corresponding expressions, please visit this paper.

Table 3 includes the average performance of the two genetic encoding systems for all selected instances. We applied the following performance metrics related to the six categories mentioned above: error ratio (ER), maximum spread (MS), generational distance (GD), inverted generational distance (IGD), spacing (S), and hyperarea and ratio (HAR).

ER is a cardinality metric that measures the proportion of solutions obtained in the non-dominated set and solutions existing both in the reference and approximation Pareto fronts. The lower values represent the better set of non-dominated solutions. According to the performance of the two encoding systems, MFBG outperforms GGA in all instances, except Gamma18, on average.

MS is a spread metric that measures the approximation's coverage across the reference Pareto front by considering the range of solutions in each objective. Higher values show better performance. The MFBG achieves better than the GGA for the instances Gamma2 and Gamma11. In the rest of the instances, the GGA outperforms the MFBG. These results show that the approximation set obtained by the GGA covers the area of reference Pareto frontier better than the MFBG in most instances.

Table 3 Average performance of two encoding systems in selected performance metrics for all test instances

Instance		*ER*	*MS*	*GD*	*IGD*	*S*	*HAR*
Gamma2	GGA	0.988	0.435	1.672	3.629	5.282	*0.917*
	MFBG	*0.965*	*0.607*	*1.636*	*3.846*	*4.721*	0.906
Gamma3	GGA	1	*1.599*	3.900	11.190	*2.617*	0.794
	MFBG	*0.982*	1.334	*3.889*	*8.402*	2.836	*0.847*
Gamma8	GGA	1	*0.532*	2.620	4.748	*1.026*	0.795
	MFBG	*0.925*	0.421	*0.909*	*2.107*	2.568	*0.904*
Gamma9	GGA	0.997	*0.564*	0.699	3.415	2.097	0.913
	MFBG	*0.985*	0.384	*0.177*	*2.325*	*0.275*	*0.941*
Gamma10	GGA	0.997	*0.474*	2.001	17.055	*2.321*	0.817
	MFBG	*0.994*	0.299	*1.227*	*12.758*	2.832	*0.873*
Gamma11	GGA	0.977	0.324	*0.540*	*3.128*	*0.989*	*0.878*
	MFBG	*0.969*	*0.431*	0.891	3.391	1.349	*0.876*
Gamma14	GGA	0.98	*0.443*	0.501	1.638	2.996	0.911
	MFBG	*0.925*	0.412	*0.223*	*1.605*	*1.178*	*0.928*
Gamma18	GGA	*0.934*	*0.393*	1.105	3.761	6.018	0.803
	MFBG	0.957	0.302	*0.218*	*3.152*	*1.131*	*0.877*
Gamma28	GGA	1	*0.481*	0.95	2.149	2.992	0.896
	MFBG	*0.931*	0.419	*0.457*	*1.706*	*0.634*	*0.921*
Gamma49	GGA	1	*2.903*	2.088	16.014	*1.521*	0.809
	MFBG	*0.991*	1.901	*1.628*	*10.441*	2.779	*0.876*

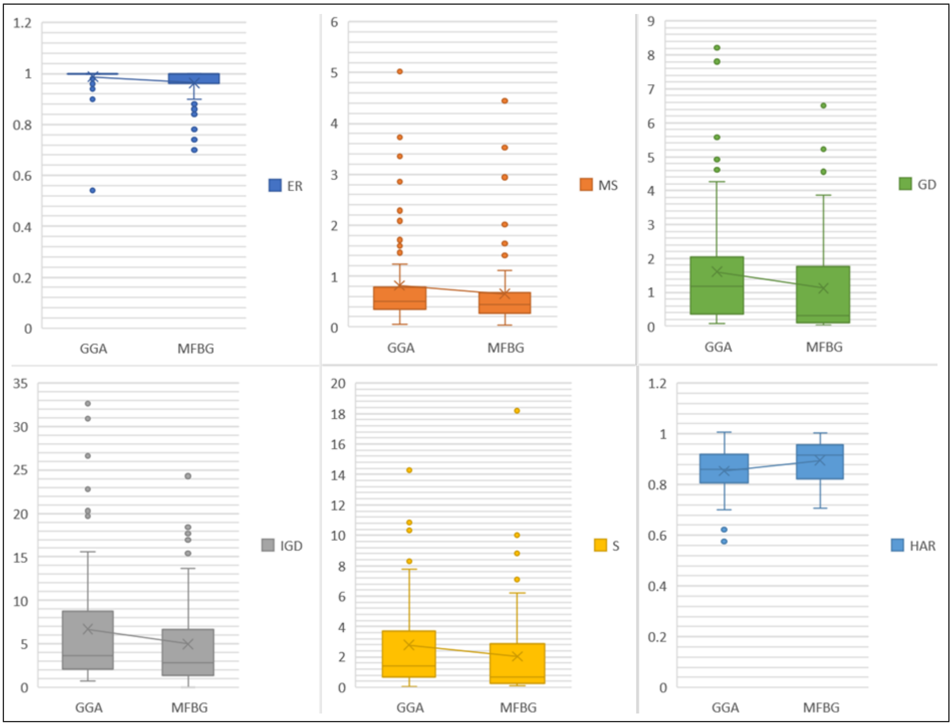
Moreover, GD is a convergence metric that measures the minimum distance between each solution that appeared on the approximation and the reference Pareto fronts. It aims at how far the approximation set is from the reference set of solutions. The lower values represent better performance. As is seen, the MFBG shows better convergence to the reference Pareto frontier than the GGA in all instances except Gamma11.

IGD is a metric that measures both convergence and the distribution of the Pareto frontier solutions. IGD estimates the distribution of the solutions in the objective space and the distance of these solutions from the reference Pareto front. Lower values of IGD show better achievement. The results occurred parallel to GD. Thus, the MFBG outperforms the GGA in all instances except the Gamma11.

S is a distribution and spread metric. S captures variations of the distance between the solutions located in the approximation Pareto front. The lower values represent better results. While the MFBG outperforms the GGA in Gamma2, Gamma9, Gamma14, Gamma18 and Gamma28, the GGA performs better in the other instances.

Finally, HAR is another convergence and distribution metric. HAR is the ratio between the hypervolume of the objective space between approximation and reference Pareto fronts. The hypervolume of the objective space refers to the area between the non-dominated solutions and a reference point. HAR represents a better performance when this ratio approaches one. According to the results presented in Table 3, MFBG shows better performance on average than GGA for HAR for all instances except Gamma2 and Gamma11. Moreover, both genetic encoding systems are performed equally in Gamma11.

Figure 7 Comparison between Falkenauer's GGA encoding system and MFBG (see online version for colours)



These results show that the MFBG performs better than the GGA in 'cardinality', 'convergence', and 'convergence and distribution' metrics. On the other hand, the GGA is better at 'spreading' metric. Both encoding systems performed well for the distribution and spreading' metric. The decision-makers can follow one of these encoding systems according to what they value more regarding the performance of their algorithm. By accessing the link https://drive.inesctec.pt/s/EXDjcB8yRpP2FN7, the reader may observe the performance of each run separately and the summary tables for each performance metric.

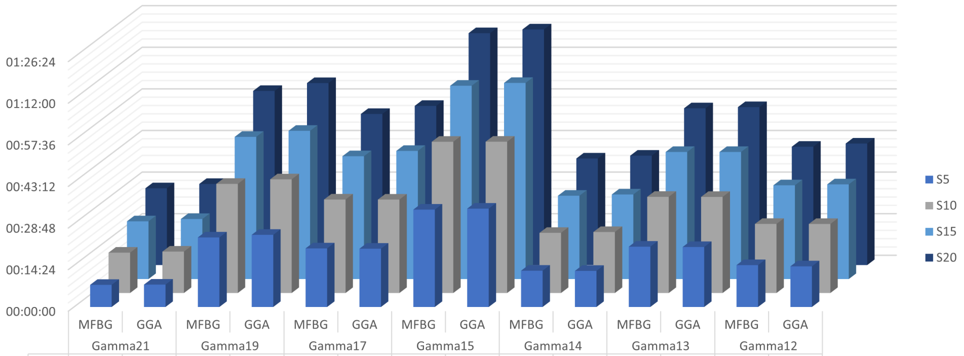
Figure 7 compares the results of the two genetic encoding systems over several trials. These box plots show the average tendency of each performance metric for the selected instances. They may provide a better and more detailed look at the results in Table 3.

We also tested the performance of the MFBG and the GGA regarding computation time (CPU time). Paired t-test statistics are used to determine the difference between the computational times. We first ran all 100 instances using the same parameters and got the CPU times. Then, under the hypothesis: mean(diff) = 0, where mean(diff) = mean(GGA_{CPU} – $MFBG_{CPU}$). We reject the null hypothesis with a p-value of 0.0369. Moreover, hypothesis mean(diff) $\leq$ 0 is rejected with a p-value of 0.0184.

This result shows that the MFBG runs faster than the GGA algorithm under the same conditions.

Figure 8 illustrates some sample results in terms of CPU times. We select seven instances, all middle size. Furthermore, we run these instances for different sector sizes, namely 5 (S5), 10 (S10), 15 (S15) and 20 (S20) for 1,000 generations. Observing CPU times is possible for both encoding systems and sector sizes in the table beneath Figure 8. As is seen, the MFBG genetic encoding system works computationally faster than the GGA for the selected sample.

Figure 8 Comparison of CPU times for different sector sizes (see online version for colours)



	Gamma21		Gamma19		Gamma17		Gamma15		Gamma14		Gamma13		Gamma12	
	MFBG	GGA	MFBG	GGA	MFBG	GGA	MFBG	GGA	MFBG	GGA	MFBG	GGA	MFBG	GGA
S5	00:07:3	00:07:4	00:24:0	00:24:5	00:20:1	00:20:1	00:33:4	00:34:0	00:12:3	00:12:2	00:20:5	00:20:4	00:14:2	00:14:0
S10	00:13:5	00:14:2	00:37:4	00:39:2	00:32:2	00:32:2	00:52:2	00:52:2	00:20:4	00:21:0	00:33:1	00:33:1	00:23:5	00:23:5
S15	00:19:5	00:20:4	00:49:1	00:51:1	00:42:3	00:44:2	01:06:5	01:07:5	00:28:5	00:29:1	00:43:5	00:43:5	00:32:2	00:32:4
S20	00:26:3	00:28:0	01:00:1	01:02:5	00:52:1	00:55:0	01:20:1	01:21:3	00:36:5	00:37:5	00:54:1	00:54:4	00:40:5	00:42:0

These results show that the MFBG performs well compared to GGA, which is one of the most used genetic encoding systems when sectorisation, assignment or grouping problems are considered. The MFBG is an excellent alternative and a contribution to the literature in the area.

6 Conclusions

This paper addressed sectorisation problems. This multidisciplinary research area is entirely relevant due to the various application fields, such as schooling, health, air traffic, political districting and location-routing. These problems, involving multiple criteria, have the desire to simplify or partition a large set or region into smaller parts, the sectors, respecting different constraints.

Various methods have been described in the literature to deal with sectorisation problems. We resorted to an approach based on GA and used the well-known algorithm NSGA-II. This algorithm is a commonly used method to solve combinatorial optimisation problems such as sectorisation, assignment, allocation or clustering. Practising NSGA-II method regarded three commonly used sectorisation criteria: equilibrium, compactness and contiguity.

We proposed a new genetic encoding system, the MFBG, expecting a noteworthy advantage in the modelling and solution procedures. This encoding system has a two-dimensional matrix form. The rows show the number of nodes, whereas the columns represent the number of sectors the nodes assign. The MFBG is subject to two strict restrictions:

1 a node cannot be assigned more than one sector

2 each sector has to be composed of at least one node.

Thus, the matrix has a binary form. The assignment status of each node is represented by filling a single cell with digit 1 in each row.

Although following existing measurements of the three criteria mentioned above, we accomplished some adequate adjustments for the approach.

Since evaluating and ensuring whether the proposed MFBG encoding system is effective in sectorisation applications was essential, we proceeded with extensive computational tests. However, the task was not straightforward due to the scarcity of data, leading us to create and make new extensive data available. Hence, 100 instances were generated (available online for access) using the gamma and normal distributions covering each node's coordinates, quantities, and contiguity maps. The smallest instance generated had 25 nodes, while the largest instance had 1,000 nodes. This instances diversity offered the chance to observe the wide-ranging behaviour of the algorithm.

The instances were clustered as small, medium and large after the number of nodes. Sector numbers were predefined according to these clusters to examine the behaviour of the proposed MFBG, within the NSGA-II algorithm, in different cases. Sector sizes were opted {5, 10} for small instances, {5, 10, 15, 20} for medium instances and {10, 20, 30, 50} for large instances.

The other parameters necessary for applying GA – the size of the population, parameters for the stopping criterion and the mutation rate – are kept constant. Given the three criteria, the population is decided as 50 to handle the complexity of the NSGA-II model. Following the various experiments, we concluded that a low mutation rate converges to a superior plateau while searching for a better result.

The effort was on finding the best possible solution, using the MFBG genetic encoding system through the NSGA-II method, separately for each objective of each criterion. Although obtaining feasible solutions simultaneously for three criteria was challenging, the algorithm attained good results in the final solution sets of distinct instances, in general. For example, the best possible case for the contiguity criterion (fully connected sectors) was achieved for almost all instances (small, medium and large) of varying sector sizes. Likewise, very favourable results for equilibrium (almost entirely balanced sectors) were obtained for all small, medium and large instances. Also, the compactness values were generally satisfactory. Consequently, several experiments showed that the MFBG genetic encoding system works in various types of sectorisation problems and can obtain good quality solutions.

Moreover, we evaluated the performance quality of the MFBG by comparing it with one of the most used genetic encoding systems proper for sectorisation, assignment and grouping problems, called the group-oriented encoding scheme proposed by Falkenauer (1992) under the GGA. We used six performance metrics and the CPU times. According to the results, the MFBG outperformed the GGA in 'convergence', 'cardinality', and 'convergence and distribution' metrics, while the GGA showed better performance in terms of the 'spreading' metric. Both encoding systems behave equally well in the 'distribution and spreading' metric. Furthermore, paired t-test statistics showed that the MFBG had lower CPU times than the GGA under the same conditions.

The computational effort of the MFBG might be affected by the two restrictions of the proposed encoding system, since infeasible solutions may appear during the crossover and mutation steps. We control the infeasibility of the solutions simultaneously by repeating the current step until a proper solution appears.

In conclusion, the approach implemented solves the multi-objective optimisation problems and the proposed MFBG genetic encoding system contributes to satisfying outcomes for small and big instances, within a reasonable computation time. The MFBG may provide an easy and clear framework for decision-makers dealing with sectorisation and its applications in every field.

Acknowledgements

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation – COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT – Fundação para a Ciência e a Tecnologia within the project 'POCI-01-0145-FEDER-031671'.

References

Agustín-Blas, L.E., Salcedo-Sanz, S., Jiménez-Fernández, S., Carro-Calvo, L., Del Ser, J., Portilla-Figueras, J.A. et al. (2012) 'A new grouping genetic algorithm for clustering problems', *Expert Systems with Applications*, Vol. 39, No. 10, pp.9695–9703.

Agustín-Blas, L.E., Salcedo-Sanz, S., Ortiz-García, E.G., Portilla-Figueras, A. and Pérez-Bellido, Á.M. (2009) 'A hybrid grouping genetic algorithm for assigning students to preferred laboratory groups', *Expert Systems with Applications*, Vol. 36, No. 3, pp.7234–7241.

Alvisi, S. and Franchini, M. (2014) 'A heuristic procedure for the automatic creation of district metered areas in water distribution systems', *Urban Water Journal*, Vol. 11, No. 2, pp.137–159.

Audet, C., Bigeon, J., Cartier, D., Le Digabel, S. and Salomon, L. (2020) 'Performance indicators in multiobjective optimization', *European Journal of Operational Research*, Vol. 292, No. 2, pp.397–422.

Bergey, P.K., Ragsdale, C.T. and Hoskote, M. (2003) 'A decision support system for the electrical power districting problem', *Decision Support Systems*, Vol. 36, No. 1, pp.1–17.

Bouzarth, E.L., Forrester, R., Hutson, K.R. and Reddoch, L. (2018) 'Assigning students to schools to minimize both transportation costs and socioeconomic variation between schools', *Socio-Economic Planning Sciences*, Vol. 64, pp.1–8.

Bozkaya, B., Erkut, E. and Laporte, G. (2003) 'A tabu search heuristic and adaptive memory procedure for political districting', *European Journal of Operational Research*, Vol. 144, No. 1, pp.12–26.

Brown, E.C. and Vroblefski, M. (2004) 'A grouping genetic algorithm for the microcell sectorization problem', *Engineering Applications of Artificial Intelligence*, Vol. 17, No. 6, pp.589–598.

Bulka, B., Carr, R., Jordan, E., Rheingans, P. et al. (2007) 'Heuristic search and information visualization methods for school redistricting', *AI Magazine*, Vol. 28, No. 3, pp.59–59.

Camacho-Collados, M., Liberatore, F. and Angulo, J.M. (2015) 'A multi-criteria police districting problem for the efficient and effective design of patrol sector', *European Journal of Operational Research*, Vol. 246, No. 2, pp.674–684.

Chen, Y., Elliot, M. and Sakshaug, J. (2016) 'A genetic algorithm approach to synthetic data production', in *Proceedings of the 1st International Workshop on AI for Privacy and Security*, ACM, p.13.

Chou, C-I. and Li, S-P. (2007) 'Spin systems and political districting problem', *Journal of Magnetism and Magnetic Materials*, Vol. 310, No. 2, pp.2889–2891.

Cotta, C., Aldana, J., Nebro, A.J. and Troya, J.M. (1995) 'Hybridizing genetic algorithms with branch and bound techniques for the resolution of the TSP', in *Artificial Neural Nets and Genetic Algorithms*, pp.277–280, Springer, Vienna.

D'Amico, S.J., Wang, S-J., Batta, R. and Rump, C.M. (2002) 'A simulated annealing approach to police district design', *Computers & Operations Research*, Vol. 29, No. 6, pp.667–684.

De Assis, L.S., Franca, P.M. and Usberti, F.L. (2014) 'A redistricting problem applied to meter reading in power distribution networks', *Computers & Operations Research*, Vol. 41, pp.65–75.

Deb, K. (2001) *Multi-Objective Optimization using Evolutionary Algorithms*, Vol. 16, John Wiley & Sons, West Sussex, England.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002) 'A fast and elitist multiobjective genetic algorithm: NSGA-II', *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp.182–197.

Degtyarev, O., Minaenko, V. and Orekhov, M. (2009) 'Solution of sectorization problems for an air traffic control area. I. Basic principles and questions of airspace sectorization and its formalization as an optimization problem', *Journal of Computer and Systems Sciences International*, Vol. 48, No. 3, pp.384–400.

Falkenauer, E. (1992) 'The grouping genetic algorithms-widening the scope of the gas', *Belgian Journal of Operations Research, Statistics and Computer Science*, Vol. 33, No. 1, p.2.

Farughi, H., Mostafayi, S. and Arkat, J. (2019) 'Healthcare districting optimization using gray wolf optimizer and ant lion optimizer algorithms (case study: South Khorasan healthcare system in Iran)', *Journal of Optimization in Industrial Engineering*, Vol. 12, No. 1, pp.119–131.

Farughi, H., Tavana, M., Mostafayi, S. and Santos Arteaga, F.J. (2020) 'A novel optimization model for designing compact, balanced, and contiguous healthcare districts', *Journal of the Operational Research Society*, Vol. 71, No. 11, pp.1740–1759.

Fox, B. and McMahon, M. (1991) 'Genetic operators for sequencing problems', in *Foundations of Genetic Algorithms*, Vol. 1, pp.284–300, Elsevier, USA.

Goldberg, D.E., Korb, B., Deb, K. et al. (1989) 'Messy genetic algorithms: motivation, analysis, and first results', *Complex Systems*, Vol. 3, No. 5, pp.493–530.

Ha, M.P., Huy, P.D. and Ramachandaramurthy, V.K. (2017) 'A review of the optimal allocation of distributed generation: objectives, constraints, methods, and algorithms', *Renewable and Sustainable Energy Reviews*, Vol. 75, pp.293–312.

Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems Ann Arbor*, Vol. 1, p.975, The University of Michigan Press, Ann Arbor. USA.

Jahuira, C. and Cuadros-Vargas, E. (2003) 'Solving the tsp by mixing gas with minimal spanning tree', *Sociedad Peruana de Computacion, II-3*, pp.123–133.

Jourdan, L., Basseur, M. and Talbi, E-G. (2009) 'Hybridizing exact methods and metaheuristics: a taxonomy', *European Journal of Operational Research*, Vol. 199, No. 3, pp.620–629.

Kim, M.J. (2018) 'Multiobjective spanning tree based optimization model to political redistricting', *Spatial Information Research*, Vol. 26, No. 3, pp.317–325.

Koduru, P., Das, S., Welch, S. and Roe, J.L. (2004) 'A multi-objective GA-simplex hybrid approach for gene regulatory network models', in *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, IEEE, Vol. 2, pp.2084–2091.

Konak, A., Coit, D.W. and Smith, A.E. (2006) 'Multi-objective optimization using genetic algorithms: a tutorial', *Reliability Engineering & System Safety*, Vol. 91, No. 9, pp.992–1007.

Lara-Caballero, A., de-los Cobos-Silva, S.G., Mora-Gutiérrez, R.A., Rincón-García, E.A., Gutiérrez-Andrade, M.Á. and Lara-Velázquez, P. (2019) 'Multiobjective genetic algorithms for reinforcing equal population in congressional districts', *Mathematical Problems in Engineering*.

Lee, C.Y., Kang, H.G. and Park, T. (2002) 'Dynamic sectorization of microcells for balanced traffic in CDMA: genetic algorithms approach', *IEEE Transactions on Vehicular Technology*, Vol. 51, No. 1, pp.63–72.

Lei, H., Wang, R. and Laporte, G. (2016) 'Solving a multi-objective dynamic stochastic districting and routing problem with a co-evolutionary algorithm', *Computers & Operations Research*, Vol. 67, pp.12–24.

Liu, C. and Kroll, A. (2016) *On the Performance of Different Mutation Operators of a Subpopulation-based Genetic Algorithm for Multi-Robot Task Allocation Problems*, arXiv preprint arXiv:1606.00601.

Marianov, V. and Fresard, F. (2005) 'A procedure for the strategic planning of locations, capacities and districting of jails: application to Chile', *Journal of the Operational Research Society*, Vol. 56, No. 3, pp.244–251.

Martí, L., García, J., Berlanga, A. and Molina, J.M. (2016) 'A stopping criterion for multi-objective optimization evolutionary algorithms', *Information Sciences*, Vol. 367, pp.700–718.

Martinho, A., Alves, E., Rodrigues, A.M. and Ferreira, J.S. (2017) 'Multicriteria location-routing problems with sectorization', in *Congress of APDIO, the Portuguese Operational Research Society*, Springer, pp.215–234.

Mourão, M.C., Nunes, A.C. and Prins, C. (2009) 'Heuristic methods for the sectoring arc routing problem', *European Journal of Operational Research*, Vol. 196, No. 3, pp.856–868.

Murray, A.T. (1999) 'Spatial restrictions in harvest scheduling', *Forest Science*, Vol. 45, No. 1, pp.45–52.

Nagy, G. and Salhi, S. (2007) 'Location-routing: issues, models and methods', *European Journal of Operational Research*, Vol. 177, No. 2, pp.649–672.

Osorio, M.A., Pérez, R. and Pérez, F. (2002) 'A comparative study between vector and matrix representations of chromosomes in TSP', *Advances in Information Science an Soft Computing*.

Pirlot, M. (1996) 'General local search methods', *European Journal of Operational Research*, Vol. 92, No. 3, pp.493–511.

Prodhon, C. and Prins, C. (2014) 'A survey of recent research on location-routing problems', *European Journal of Operational Research*, Vol. 238, No. 1, pp.1–17.

Pukelsheim, F., Ricca, F., Simeone, B., Scozzari, A. and Serafini, P. (2012) 'Network flow methods for electoral systems', *Networks*, Vol. 59, No. 1, pp.73–88.

Ríos-Mercado, R.Z. and Bard, J.F. (2019) 'An exact algorithm for designing optimal districts in the collection of waste electric and electronic equipment through an improved reformulation', *European Journal of Operational Research*, Vol. 276, No. 1, pp.259–271.

Richards, E.W. and Gunn, E.A. (2003) 'Tabu search design for difficult forest management optimization problems', *Canadian Journal of Forest Research*, Vol. 33, No. 6, pp.1126–1133.

Rodrigues, A.M. and Ferreira, J.S. (2015) 'Sectors and routes in solid waste collection', in *Operational Research: IO 2013-XVI Congress of APDIO*, Springer International Publishing, Bragança, Portugal, 3–5 June, pp.353–375.

Rudenko, O. and Schoenauer, M. (2004) 'A steady performance stopping criterion for Pareto-based evolutionary algorithms', in *6th International Multi-Objective Programming and Goal Programming Conference*.

Salhi, S. (2017) *Heuristic Search: The Emerging Science of Problem Solving*, Springer, Palgrave Macmillan Cham, USA.

Sarrazin, R. and Smet, Y.D. (2016) 'Design safer and greener road projects by using a multi-objective evolutionary approach', *International Journal of Multicriteria Decision Making*, Vol. 6, No. 1, pp.14–34.

Schaffer, J.D. (1985) 'Multiple objective optimization with vector evaluated genetic algorithms', in *Proceedings of the First International Conference on Genetic Algorithms and their Applications*, Lawrence Erlbaum Associates Inc., Publishers.

Song, M. and Chen, D. (2018) 'An improved knowledge-informed NSGA-II for multi-objective land allocation (MOLA)', *Geo-Spatial Information Science*, Vol. 21, No. 4, pp.273–287.

Srinivas, N. and Deb, K. (1994) 'Muiltiobjective optimization using nondominated sorting in genetic algorithms', *Evolutionary Computation*, Vol. 2, No. 3, pp.221–248.

Tang, J., Alam, S., Lokan, C. and Abbass, H.A. (2012) 'A multi-objective approach for dynamic airspace sectorization using agent based and geometric models', *Transportation Research Part C: Emerging Technologies*, Vol. 21, No. 1, pp.89–121.

Tsai, M-W., Hong, T-P. and Lin, W-T. (2015) 'A two-dimensional genetic algorithm and its application to aircraft scheduling problem', *Mathematical Problems in Engineering*.

Vanneschi, L., Henriques, R. and Castelli, M. (2017) 'Multi-objective genetic algorithm with variable neighbourhood search for the electoral redistricting problem', *Swarm and Evolutionary Computation*, Vol. 36, pp.37–51.

Wallet, B.C., Marchette, D.J. and Solka, J.L. (1996) 'Matrix representation for genetic algorithms', in *Automatic Object Recognition VI*, Vol. 2756, ppp.206–214, International Society for Optics and Photonics.

Xie, S. and Ouyang, Y. (2016) 'Railroad caller districting with reliability, contiguity, balance, and compactness considerations', *Transportation Research Part C: Emerging Technologies*, Vol. 73, pp.65–76.

Yanık, S., Kalcsics, J., Nickel, S. and Bozkaya, B. (2019) 'A multi-period multi-criteria districting problem applied to primary care scheme with gradual assignment', *International Transactions in Operational Research*, Vol. 26, No. 5, pp.1676–1697.

Yazdandoost, F. and Izadi, A. (2016) 'A decision-making framework for designing water distribution networks based on multi-objective optimisation', *International Journal of Multicriteria Decision Making*, Vol. 6, No. 4, pp.269–289.

Yen, G.G. and He, Z. (2013) 'Performance metric ensemble for multiobjective evolutionary algorithms', *IEEE Transactions on Evolutionary Computation*, Vol. 18, No. 1, pp.131–144.

Yin, C.W.S. and Suresh, S. (2018) 'Preference-based 3-dimensional en-route airspace sectorization', in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp.769–776.

Zhang, K., Yan, H., Zeng, H., Xin, K. and Tao, T. (2019) 'A practical multi-objective optimization sectorization method for water distribution network', *Science of the Total Environment*, Vol. 656, pp.1401–1412.

Zheng, W., Tan, Y., Fang, X. and Li, S. (2017) 'An improved moea/d with optimal de schemes for many-objective optimization problems', *Algorithms*, Vol. 10, No. 3, p.86.

Zitzler, E., Deb, K. and Thiele, L. (2000) 'Comparison of multiobjective evolutionary algorithms: empirical results', *Evolutionary Computation*, Vol. 8, No. 2, pp.173–195.

Zou, X., Cheng, P., An, B. and Song, J. (2016) 'Sectorization and configuration transition in airspace design', *Mathematical Problems in Engineering*.

Notes

1 The compactness measure presented in equation (8) is selected due to its adequacy for the data used in this paper. The average value of the distances of all basic units to the centroid can also be a good alternative to understand how dense the sectors are. However, using the method presented in equation (8), the compactness can be calculated faster than measuring the distances to the centroid of all the basic units assigned in the same sector. Decision-makers should know their data and consider adequate measurement for compactness.